

Still Alive With Sir Clive!

# ZXir QLive Alive!

The Timex/Sinclair North American User Groups Newsletter

Volume 10 No. 1

Spring 2000

## MEMORY MAP

### ADDRESS

### ROUTINES

- 2 Information and Chairmen — Treasury Note\$
- 3 *Input/Output* — by Abed Kahale
- 3 Allegations of Web Page Theft
- 8 D. G. Smith & Jeff DeCourtney
- 8 Sinclair E-Mail List
- 9 From the Chairman's Disk — Donald Lambert

### FILES

- 10 The Platform - I — by Jeff Burrell

### ADDRESS

### FILES

- 14 **QL Hacker's Journal**
- 18 Diskette Menu TS-2068 — David Solly
- 21 TS-1000 I/O Board — Joe Rampolla

### SUBROUTINES

- 23 Unclassified Ads



ZXir QLive Alive! ©

Established 1991 The Timex/Sinclair North American User Groups Newsletter

# T/SNUG Information

We wish to support the following platforms : ZX-80/81, TS-1000, Spectrum, TS-2068, Z88 and QL. If you have any questions about any of these fine Sinclairs, contact the:

## Chairman

Chief Motivator  
Donald S. Lambert  
738 Gunnar Ln.  
Forsyth, IL 62535  
(217) 875-8043

## Vice-Chairmen

**Z88 Library**  
Dave Bennett (HATSUG)  
1275 Timber View Dr  
Mechanicsburg, PA 17055-9146  
717 732-4374

## QL Hacker's Journal

Timothy Swenson  
2455 Medallion Dr.  
Union City, CA 94587-1914  
swensontc@geocities.com

**TS-2068**  
Rod Humphreys (VSUG)  
10984 Collins Pl  
Delta, BC V4C 7E6 Canada  
604 583-2819

**QL PD Library**  
John Donaldson (CATUG)  
835 Foxwood Cir.  
Geneva, IL 60134-1631  
630 232-6147

**AERCO & Z80 Emulator**  
Keith Watson  
41634 Amberly Dr.  
Mt. Clemens, MI 48038

**BBS ---GATOR---**  
Bob Swoger (CATUG)  
613 Parkside Cir.  
Streamwood, IL 60107-1647  
630 837-7957 Work 847 576-8068

Any of the above can also be reached by E-Mail through the  
**Club BBS 847 632-5558**

## ZXir QLive Alive!

Is the newsletter of T/SNUG, the Timex/Sinclair North American User Groups, providing news and software support to the T/S community in a VOLUME of four newsletters per year; beginning with the Spring (March) issue.

T/SNUG's main goal is to preserve and encourage the use of Sinclair computers by providing an open forum for the exchange of knowledge, building and maintaining of software libraries. Providing vendors, repair service and members with free ad space.

It is the user groups and individual subscribers, rather than the vendors, that provide the pecuniary support for this newsletter. Vendors and developers receive this newsletter free of charge, though contribution from vendors and user groups is gratefully accepted. Please support our vendors and service providers whenever possible.

If you have a problem or you have solved a problem, please share it with the rest of us. No problem will be considered unimportant.

## Editor/Treasurer Publisher

You can keep T/SNUG alive by an annual contribution of \$12 for one VOLUME made payable to Abed Kahale. Send check to:-

ABED KAHALE  
3343 S FLAT ROCK CT  
SIERRA VISTA AZ 85650-6874  
520 378-3424

Back copies are available for  
**\$1.00 each postpaid.**

## Trea\$ury Note\$

As of March 7, 2000, we have a balance of \$940.12

## Article Contributions

Send in your articles by disk, hardcopy mail, or e-mail and your inputs to—

**Abed Kahale**

E-mail: AKahale@compuserve.com

## Welcome

*Jeff Burrell*

## GATOR'S

## Twisted Pair

To better inform the Sinclair Community, three 24-hour a day BBSs are now provided to serve you. You are encouraged to exchange mail and use the files sections of these boards. Bulletins and ads are available to all

**Q-Box BBS 810 254-9878**  
Utica, Michigan

**SOL BBS 520 882-0388**  
Tucson, Arizona

**Club BBS 847 632-5558**  
Arlington Heights, Illinois

## WEBPAGES

<http://users.aol.clubbbs/tsnug/>  
<http://www.outlawnet.com/~jboatno4>

If you know the Internet E-Mail address of a Sinclair user, but do not have access to Internet, simply address your E-Mail to GATOR Sinclair on the 24-hour Club BBS and include the name and E-Mail address of the user you wish to reach. Then check the Club BBS from time to time if you expect a reply.

We encourage you to exchange mail and contribute to the UPLOAD section. Call and register using your first, last name and phone number along with a password you won't forget. **Write It Down!** Do not try to do anything else at this time.

When you call-in the next time, you will have Level 5 security and be able to enjoy full user privileges. The BBS has smaller sections called conferences. Select "J" for "Join a Conference". Select "TIMEX" to get into the Sinclair Section. The mail you then read will only be from other T/S users. Use extension .ART for articles, .ADS for ads and .NWS for news when UPLOADing.

For help, contact the SYSOP, Bob Swoger, by leaving a message, mail, E-Mail or phone.  
CENG108@email.mot.com



# Input/Output

by *Abed Kahale*

Hi again,

I'm about 45 minutes north of Denver in Longmont, Colorado. Also, is there an email for John Oliger? I would like to find out about the Safe DOS system he has. I recently got out my TS 2068 to show my 7-yr. old and he likes the games.

We mainly play Starion on the Spectrum side of the ROMSwitch. So, I figured I'd put together the whole disk system to show him what's going on. Since I work at Seagate, I was able to pick up a power supply for the disks that already have.

Anyway, sorry for rambling, I finally got the system to work and had forgotten how much software I had on disk. I also built a RAMDisk, but I believe there are some errors in either the soldering or assembly or both. I emailed Larry Kenny to see if he can assist me in repairing it. If he can't, is there someone you know that I could send the board to for evaluation and repair. Of course I would pay for the service.

I currently have a little 9" TI monitor. When I called up MScript or Tasword, the 64 columns were really hard to read. Do you know of a monitor that gives a cleaner 'picture' for the smaller typeface? I saw a lot of items for sale in the Ad section, but am not sure what to get. Any assistance would be greatly appreciated. Sinclairly,

**Paul Anderson**

I put a bid, (through the mail) on the items, after I sent the text to you. It was a lowball bid of \$50.00 for the whole thing. I didn't think that I had bid enough, and I was surprised to find out, that I had one. I've been talking to John Rish of Home Electronics Service in San Antonio, Texas. He says that, they can change the power supply over to the US electric current. I'll keep you posted. HAPPY HOLIDAYS!

**Dane Stegman**

## INTELLIGATIONS

## BEWARE

I wanted to drop you a note to let you know that there is a fellow who claims to be a TS enthusiast and tries to set up trades and sales with **unsuspecting** folks. His name is Phillip Ewing. I have been contacted by a number of people who have said that their dealings with him have been less than satisfactory. The fellow has even stolen the web page I've been working on and is calling it his own.

Take a look at:

<http://www.topcities.com/Computers/ts1000/TimexShistory.htm>

I thought I should warn you so that if you hear or are contacted by anyone about him you'll know to steer clear of him. Somehow the word should get into ZQA! So that he

doesn't cheat any of the membership. I am notifying everyone I know about this.

Take care,

**Jack Boatwright**

Rod,

Thanks so much for the help and concern. The web page is now down (at Least at that url), but I suppose it could come back as a different Address.

Anyway, I appreciate what you did to help. I had sent him 2 emails and Contacted his isp yesterday, but haven't heard back from either one.

Take care and let me know if there is anything I can do to repay your kindness,

rodh wrote:

>From: "Ewing, Phillip A" <pewing@glsp.org>

>To: "rod h" <rodh@lightspeed bc ca>

>Subject: RE: what the hell??

>Date: Mon, 3 Jan 2000 12:00:20 -0500

>It is off.

This guy is a real loser, he's about 29, works for a law firm in Atlanta, Georgia and has made enemies with some TS folks I know. Mostly trading with them and not following through with his part. But when he took all my work, photos and information and called it his own he really pissed me off (sorry for the language)! I will not tolerate theft. His page is now down. Hopefully for good, but it could resurface with a different address.

On another note, if you are ever contacted by a fellow named Phillip Ewing please be very careful dealing with him. There are a number of people he has dealt with for Timex Sinclair items in a less than satisfactory way. He has even stolen the web page I have been working on and is calling it his own.

Did you see the ruckus about the guy in Atlanta, Philip Spivey, copying Boatwright's web page exactly and just putting his name where Boatwright's had been? Rod Humphrey really chewed him out. Then Spivey apologized like he was 15 years old when he is really 29 years old. His excuse was that nobody ever respected him.

This guy has not been a good person. He has had bad dealings with at least two T/SNUG members (Glen Goodwin and Luke Perry) besides trying to steal the web page. There are others (non-members) who have contacted me as well.

Please be careful in how you provide the information in ZQA!, I do not want to say something that I (or you) could get sued for saying. Probably best not to provide names (mine is OK) unless those folks say it's OK. Just a note to warn everyone of unscrupulous people lurking on the internet ready to take our money and other things.

Please take care in dealing with folks on the internet, be cautious and know who you are dealing with. Recently, the Timex Sinclair web page I have been working on was completely stolen and re-advertised as someone else's work. I also found out that the person who did this had also been trading with people for Timex items and not fulfilling his side of the bargain, promising something that

sounded good but never delivering, or not delivering what was promised.

As in everyday life, there are criminals lurking on the internet taking advantage of unsuspecting people for their own gain.

Here are some things to keep in mind when dealing with people on the internet:

Fraudulent sellers will often adopt multiple e-mail accounts that allow them to switch identities at will.

Criminals who prowl auction sites can use fraudulent credit cards to establish legitimate-appearing accounts.

People will promise to give you something that you want for something that you have and not follow through even though you have. Remember, if something seems too good to be true is probably is.

Jack

Good to hear Abed. I'm thinking of getting some upgrade's done on my PC. I haven't heard from John Rish, yet, about the Spectrum computer. As soon as I do, I'll let you know!! Thanks!

Dane Stegman

*Yeah! I lost my address book in the process of bigger and faster. But I still have the membership addresses in my trusty TS-2068.*

**Please put a comment in N/L asking if anyone ever heard of an emulator for TS-2068 to Mac.**

JOAN KEALY

PO BOX 1439

BRACKETTVILLE TX 78832-1439

[hjkealy@hilconet.com](mailto:hjkealy@hilconet.com)

Hi Abed,

My wife forwarded your message to me here. Anyway, I need to be my RAMDisk repaired. I emailed Larry Kenney and he was no help. I have been emailing with Jack Boatwright and he sent me the directions to build the RAMDisk. I couldn't find mine. So, I need to review the process and find where there may be a cold solder or a fracture. I've gotten my entire system set up and running again.

It took some time, but I got the cobwebs out and it works fine. Can't say the same for many older PCs. Oh well. Thanks for asking. Sinclairly,

Paul Anderson

I'm sending this mail because there's a great new service called MSN Messenger Service that will help us stay in touch. I'm hoping you will sign up so we can exchange instant messages online.

It's fast and easy to sign up, and it's free! Just go to the MSN Messenger Service web site to get more information and to install the software

Once you've installed MSN Messenger Service, add me to your list of contacts: my e-mail address is: [k\\_david\\_solly@hotmail.com](mailto:k_david_solly@hotmail.com).

(You might want to write my address down because you will need it when adding me to your list.) I look forward to sending you an instant message soon!

Note: If the link above doesn't work, copy and paste the following URL into your browser:

<http://messenger.msn.com/lcen/friend/default.asp?userema>

ZXir QLive Alive!

[il=k\\_david\\_solly@hotmail.com](mailto:k_david_solly@hotmail.com)&username=David

Just letting you know of our new email address. Since Prodigy took away our local access number, we switched our ISP. Anyway, please note that there isn't any 'o' in the last name. Our new ISP is called Peak-to-Peak; hence the 'peakpeak'. So, please send your emails to this address since we will be canceling our Prodigy account this week. Thanks for your support.

Also, I know that some of you will receive duplicates of this message due to multiple addresses in our list. I apologize. I just selected all addresses to send this message. Take Care,

Paul and Debbie Anderson

[pandersn@peakpeak.com](mailto:pandersn@peakpeak.com)



**People**

**ropping**

er places got their  
n facts, legends  
aginations.

Odds are you'll look twice at the city limits sign near Tom and Sharon Kidd's Kentucky home.

"Legend says residents planned to name the town Mt. Washington, but the Post Office said that there were enough Washingtons in the nation," Sharon says. "To make sure their second application was accepted, they proposed something more uncommon."

The Kidds' daughter Jennifer points out the peculiar outcome above.

Abed. I mailed my check for another year's subscription to ZXir QLive! Newsletter. I've shipped my Spectrum +2 to John R. Rish of Home Electronics Service, San Antonio, TX, last week. He'll let me know, what I need to get it up and running. I'll keep you posted. HAPPY NEW YEAR 2000!!!

Dane Stegman

Firstly I found you name & address in a file on Tony Firshman's BBS, QBBS, called QLusers.txt & I was wondering if you could help me?

I used to use my Sinclair QL extensively from 1984 through to 1994, when fate allowed me to afford a PC. Since then I have had my original QL, ExpanderAM, Trey Card & disk drives <STOLEN>. Consequently I no longer have access to my copy of Toolkit II.

I have now discovered QLAY emulator to run under Windoze, so I'm please to return to work on programs I developed a long time ago.

Although QLAY can use the image of the Minerva ROM, which is a good thing, as I need PIPE's, SElecting Strings \* Integers, etc, I still need an image of the Toolkit II ROM, that was in the disk interface. My programs use many features of the Toolkit, as well as Minerva features, such as BPUT, BGET, etc.

If you could send me copy of the Toolkit II ROM I

could then use this with QLayer & the Minerva ROM & then continue to develop my old SuperBASIC programs TIA  
Regards

Tel: +44 705 069 5844 - Fax +44 705 069 5845

Email Address: QL-user@cslewu.free-online.co.uk

IP Address: ppp-3-69.cvx4.telinco.net

Using Aureate Group Mail Free Edition

<http://www.group-mail.com/>

**Andy Barber**

*I am forwarding this message to one of our members,  
Ai Feng who is very well versed with the QL.*

Sender: QL-user@cslewu.free-online.co.uk

Thanks for the reply, the problems is now sorted.  
Regards,

**Andy Barber**

Sender: P.Liebert@t-online.de

Please tell it to all your friends: ZX-81 users meeting 17 - 19 march 2000 in Germany. Everyone who loves SINCLAIR ZX80, ZX81, TIMEX TS1000, TS1500, JUPITER ACE, PC8300, POWER 3000 is invited to participate: It's the 4th big meeting of German ZX81 users group ZX-TEAM.

For more information visit my updated homepage: <[www.zx81.de](http://www.zx81.de)>

ZX-TEAM-Homepage: <http://www.zx81.de>

ZX81-Web-Ring [http://home.t-](http://home.t-online.de/home/sinclair_zx81/zx81_wbr.htm)

[online.de/home/sinclair\\_zx81/zx81\\_wbr.htm](http://home.t-online.de/home/sinclair_zx81/zx81_wbr.htm)

**Peter Liebert-Adelt**

I put a bid, (through the mail) on the items, after I sent the text to you. It was a lowball bid of \$50.00 for the whole thing. I didn't think that I had bid enough, and I was surprised to find out, that I had one. I've been talking to John Rish of Home Electronics Service in San Antonio, Texas. He says that, they can change the power supply over to the US electric current. I'll keep you posted.

**Dane Stegman**

Hi Abed,

Got the Winter edition of the newsletter today. Thanks. Thanks for running my request for a T/S2068 Microdrive. I did end up getting a LarKen/Oliger disk setup through Jack Boatwright along with some software. I just thought I would add something that you might want to run in the next issue.

There is a fellow in my area (Portland, Oregon) that does electrical repair on electronics (computers, audio, etc.) I guess he used to work for Intel as an engineer and he really knows his stuff. Anyway, he is able and willing to do repair work on old Timex/Sinclair gear as long as he has a schematic and spare parts (or if the parts are available locally). I was thinking now that since "Computer Classics" is out of business, this might be an alternative. If people are in need of T/S repair work they can contact me and I can run it by him, or they could always call him directly. His number and address are given below:

**Bill Mathis**

**Pro Digital Electronics**

**323 SE 28th Ave.**

**Portland, Oregon 97214**

**503-232-3200**

**Luke Perry**

I am working on an article or column for ZQA on how to get the Z88 to talk and listen to the IBM PC. The Compaq refers to things attached on the back as TWAIN devices and the Z88 refers to them as WEIRDOS. So I am trying to marry a *weirdo* to a TWAIN and I am almost afraid of what each computer will complain about. I have progressed far enough that I am ready to try it. I have the cable between set up and the software for the Z88 is on a plug in EPROM and the software is loaded for the Compaq. I had to get an adapter for the 25 pin serial cable to the 9 pin the Compaq has. I know it can be done so that is a good point in my favor.

To load the program in to the Compaq I had to get into DOS. The Compaq has DOS 7.0 and the DOS manual that I have is 5.0 but I got that done after I realized that the exact directions were in the PC-LINK User Guide.

**Don Lambert**

Well, Abed, he charged me \$25 for a half hour of time, to replace my defective floppy drive with a used one. The part cost \$2 plus state sales tax.

I report this, just to let the Sinclair community know, that there is a guy out here, who will patiently pick apart a customized piece of equipment, in order to diagnose and repair it. And, you can see how he charges!

His name is "Jon", and he can be reached below:

**Pro Act Consulting, Inc.**

**2660 N. Houghton Rd.**

**Tucson, AZ 85749**

**520-749-5395, fax 520-749-3626**

**email <[proactmd@aol.com](mailto:proactmd@aol.com)>**

Well, the little printer finally gave up the ghost!

So, I'm in the market.

**Do you know where I can  
get a 2040 printer  
esp. with an extended cable**

(in order to better fit into my configuration.)

Sysop, SOL BBS @ 520-882-0388

520-882-3972 (voice)

[emanon@azstarnet.com](mailto:emanon@azstarnet.com) (email)

**David E. Lasso**

**2590 N. Jordan DR**

**Tucson AZ 85745-1132**

Dear Abed,

The other day while sorting a few boxes of stuff, I ran across a **printer IF & cable** which I didn't know I had. As I recall, it's for connecting a Timex 2068 to a dot matrix printer. I have no further use for the cable/IF plus **two Epson dot matrix printers** (a 9 pin LX-800 with a non-tractor feed and a tractor feed and a box of tractor feed paper). The other printer is a 24 pin 3250 Epson. I also have quite a few used spare ribbons, for both printers, which I had been "re-inking" with mineral oil.

**I'd be willing to TRADE  
or SELL ANY or ALL of the  
above for the best offer.**

If anyone is interested, I can be contacted at: Email:

[oranur@juno.com](mailto:oranur@juno.com)

Phone: (716) 691-9495

**Fred Henn**

**230 N. French Rd.**

**Amherst, NY 14228-2033**



(My hobbies are Electronic organs, Electronics, Computers, Music, and Gardening).

I recently made contact with our "old" local Timex Tech (Conrad Zaranski) and I've had a fair amount of Email correspondence with the local gentleman (Dane Stegman) whose name & Email address you kindly furnished awhile ago. Dane recently "picked-up" a Sinclair Spectrum computer

BTW, Abed, have you heard anything more on Uncle Clive's new laptop (which he was to start marketing this year)? Sinclairly,

**Fred Henn**

oranur@juno.com

Thanks for the back issues: they were informative and fun to read. The adverts alone are worth the price of admission!

1. I am working at the interface as I get time.
2. I have finished a semi-commented disassembly of the 2068 home ROM and EXROM. I have assembled the result and found them to be byte-for-byte identical to the ROM images I downloaded from the net.
3. I have been working on a Windows based debugger for checking my code.

**Jeff Burrell**

JBurrell@endocardial.com

Hi Abed,

I have been looking over my T/S 1000 program for that Byte-Back I/O board and I was wondering if you know if a BASIC compiler was still around for the old Timex. I don't know if the PAUSE statement would be the same if the BASIC program was compiled into machine language, but I never got into the compilers. Don't go to any trouble about it, but I thought I'd ask if you might know, off the top of your head.

I also was wondering if that Byte-Back I/O board works in FAST mode. The Byte-Back modem was sensitive to FAST and SLOW mode and would sometimes crash. I don't want to try it since I don't want to damage anything on that I/O board. Could you also ask if anyone else is using that BB-1 I/O board by Byte-Back if it isn't too much trouble? Take care,

**Joe Rampolla**

jprampolla@blazenet.net

Thanks again for Mr. Liebert-Adelt's address. I will keep you posted!

Hi Abed, in regards to a TS2068 disk drive interface, I believe Jack already got rid of the one that he had. I know it is kinda late in the game to be looking for one of these things but I thought it would be worth a try. Maybe you could do me a favor and add in the newsletter that I am also looking for a **microdrive system** for the TS2068, I believe it was made by A&J. I sure would appreciate it! I am looking forward to my first copy of the newsletter. Thanks for all of your help Abed.

**Luke Perry**

The RAM and ROM are presently mapped into the dock space using the ROSCS from the 2068 connector (I have not been able to find a source to tell me how to use the bank selection logic in the 2068 CPLD). At any rate, I am presently fixing the self-inflicted wounds caused by wiring errors (yes, it's point-to-point hand wired [weird?]) and design errors. I should have the hardware wrung out by the first of the year. It will take this long because I am doing this in my "spare" spare time.

Would the T/SNUG be interested in some written information on this project. I can submit it in PDF format. I can include schematics and a description of the design and updates as available on the drivers.

I am also in the process of disassembling and commenting the system ROMs for the 2068. Is there a source of commented code out there? I have the Spectrum disassembly, but there are enough differences (and the comments are in German) that it is more trouble than it is worth. I am using Ian Logan's disassembly of the ZX81 as a guide.

**Jeff Burrell**

JBurrell@endocardial.com

Please send ZQA! To:

João Paulo Vaqueiro Encarnado  
Rua do Castelo, Nº24 R/C Esquerdo  
Pirescoxe  
2695-247 Stª Iria de Azoia  
Portugal

On the internet he goes by the name Johnny Red and has a web site called "Timex Computer World" which is linked on the bottom of the page at my site. Take care,

**Jack boatwright**

Hi Abed,

Got a nice reply from Peter, and seems like he might be able to answer a few of my questions. Just like I suspected, I might have a few problems to work out before I could use the compiler, but at least I am on the right track. Thanks again for the help! Take care,

**Joe Rampolla**

The three pages of schematics I sent are all for the interface board. I am currently using the Altera 7128 for glue logic but have just completed the design to use an Altera 10K10 FPGA for the next go if anyone else is interested. If there is no further interest in the interface among the 2068 group I will probably finish it up the 7128 version for my own education and then press on to building a solid-state disk for my Z88 using the Toshiba parts and a PIC 17C42 or 17C44 processor as I mentioned in my first article. I am planning to make the SSD compatible with the Tandy portable drive spec so that current Z88 software could use it.

By the way, does anyone still have QLs to sell? I would like to have a complete stable of the main Sinclair US machines.

**Jeff Burrell**

4955 Emerson Ave N  
Minneapolis, MN 55430  
JBurrell@endocardial.com

I have been in contact with John Rish of Home Electronic Service regarding the Spectrum +2 computer. I will keep you posted. Thanks.

**Dane Stegman**

26 Marshall Ave.  
Akron, NY 14001

I have been studying the relationship of film photography and digital photography and that is a go from circles of confusion (that is a technical term relating to the sharpness of an image on film) to pixels. While they are terms that mean about the same in the type of photography they refer to they cannot be used with the other type directly since they measure different things.

They both have something to do with resolution of the resultant picture but go at it a different way. Actually,

pixels are more like the grain size of the silver compounds than anything else. I went to a photographic school in Dallas, TX way back in 1948-49 but I never followed up in that career. Nor did I try to keep up with the advancement of the technology until the last few months and I am trying to learn more about marrying a camera to a computer by way of digital cameras. I don't know if I want to spend umpteen bucks for a camera that has less resolution than a cheap point-and-hoot camera

**Don Lambert**

My poor eyesight doesn't allow me to use my TS equipment very much, but still enjoy every issue of T/SNUG very much. Keep up the good work.

**Earl Kielglass**

Here is my annual contribution of \$12, the other \$8 you can put in the treasury. I was reading about David Lasso's complaint about Computer Classics and his disk having problems but my problem, I think, is the LarKen interface. I can not get any of my disks to RUN the drives. It START RUN a couple of seconds. I have 5 drives and they all do the same thing. The 2068 runs good with the tape drive, would like to get the LarKen interface fixed if anyone you know fixes them. 73es.

**Harry Miller**

16 West St.

PO Box 62

Berlin, MA 01503

I am working apace at the interface, but haven't had much time since the first of the year

PS Put the extra in the treasury.

**Jeff Burrell**

Hello Abed, please find my check for membership to ZQA! Newsletter. I am looking forward to another year of membership. I do have a question of you if I might... How do I go about finding what software the group has in its public domain library? I did email someone whose name was listed in the newsletter, but that was a couple months back and I have yet to hear a reply.

Thanks Abed and thanks for all of the work you put into keeping the Timex and Sinclair computers Alive!

**Luke Perry**

<doidy34@yahoo.com>

809 J. Q. Adams 557-9978

Oregon City, OR 97045

*I appreciate your kindness, thank you*

Regarding your asking if I am looking for ZX81/TS1000 software...were you going to put in a request in ZQA! for me? What I am really looking for if you could include it in the next issue, is a

## **modem**

for my TS2068 (I believe it is the **TS-2050**). I would really appreciate it if you could include that. Thanks Abed and here is my address:

**Luke Perry**

3708 NE 109th Ave. #115

Vancouver, WA 98682

doidy34@yahoo.com

Your site was recently submitted to OpenHere.com. We have reviewed your site and decided to include it as follows:

Link: <http://users.aol.com/clubbbs/tsnug/>

Title: Timex/Sinclair NorthAmerican User Groups

ZXir QLive Alive!

Description. OpenHere Category:

<http://www.openhere.com/tech1/software/operating-systems/sinclair/organisations/>

As you are listed as a contact person on the home page of this site, I am dropping you a quick note to let you know about your inclusion on OpenHere.

OpenHere is one of the 10 largest index and search sites on the Internet and is specifically focused on creating resources for the family.

You can dynamically modify your site's listing at any time, or include your site's listing in other categories on OpenHere.com.

When you modify your site's listing, it is automatically placed at the top of the category in which it is included, and is placed first in the search engine results for the keywords relating to your site.

To modify, add or delete your listing:

1. Go to the OpenHere category where your site is listed.
2. Click on the "Suggest a Site" link.
3. Follow the instructions for changing your listing

All of the modifications you submit to OpenHere.com are processed in real time. As soon as you see the response to your submission, your site listing should be updated.

OpenHere is frequented by both children and families. As a result, www.OpenHere.com does not include links to material which is illegal to display to minors.

If you have a question, or need help in any way, please just send me a note.

Yes, I am a real person! Sara [www.OpenHere.com](http://www.OpenHere.com)

**Don Lambert**

[dslambert@compaq.net](mailto:dslambert@compaq.net)

Abed,

Now sometimes the sound doesn't work in the 2068, but the problem is intermittent! For example, sometimes, when I use Jack Dohany's DFM, it BEEPS upon loading itself. But, just now, it does not BEEP.

Oh, BTW, the sound generator works, but the BEEP is not working right. I expect no problems, playing some of Joan's programmed songs

! KEEP ON TIME X'n !

**David E. Lasso**

*Most likely it is the sound chip pins not making good contact after so many years. Your best bet would be if you know of someone who could remove the chip and then re-install it into the socket.*

I only have enough Spectrum EPROM's for a project I've been working on with Alvin Albrecht. Alvin designed, and I have been putting together, some Spectrum emulator cartridges for plugging into the 2068 dock port. I still need to solder a part on each board and test them. I'm not even certain that the LarKen version of the EPROM's will work on these boards yet. Take care,

**Jack Boatwright**

Sir

You inquired as to keeping the John Oliger Co. ad in ZXir QLive Alive? Yes, please do! I get very few orders for TS2068 products nowadays, but still carry pretty much everything for this computer.

Thank you for the support!!!!

**John L. Oliger**

[joliger@mindspring.com](mailto:joliger@mindspring.com)

Dear Abed,

We thought you'd like to know that **Jeff (DeCourtney, CATUG)** passed away November 29, 1999. His death was caused by double pneumonia. He was only 47.

Jeff was always grateful of the extensive help that you gave him T/Sing. Before he died he had worked up to a 486 computer, had taken courses in Windows, etc., and used e-mail extensively, especially with relatives.

We miss his good natured presence very much. He was Lois' only son and had lived with or near her all of his life. He suffered from a variety of health problems throughout his life courageously and perhaps his passing is a blessing although his loved ones, and especially his mother, can never get over the loss of one of her children.

We thank you for all of the time and attention that you gave to Jeff. May God bless you and yours. Sincerely,

Lois (mother of Jeff) and George (stepfather) Plondke



**D. G. Smith**

Vice Chairman

Tape & JLO PD Library

The last issue of ZQA!, Winter 99, to Denise Smith was returned marked - Deceased

## SINCLAIR E-MAIL LIST

Anderson, Paul	pandersn@peakpeak.com
Anson, Gerald	jerrya@aztec.asu.edu
Barker Robin	robin@di-ren.demon.co.uk
Bennett, Dave	dbennett10@desupernet.net
Boatwright, Jack	jboatno4@outlawnet.com
Boehm, Al	boehm@ziplink.net
Boehm, Bill	boehm@plh.af.mil
Burt, Richard	ajb@intranet.ca
C. A. T. S.	mf0002@epfl2.epflbalto.org
Castro Antonio	castrox@portoweb.com.br
Catotti, Christopher	kd4ace@compuserve.com
Chambers, George	gfchamb@pathcom.com
Collins, Bill	bcollins@home.ifx.net
Cottrell, Les	jacottre@gte.net
Cruz-Figueroa, Jaime	cruzfiguer@aol.com
Dansby, Andrew	adansby@atlantic.net
Davis, Frank	fdavis@iquest.net
Dunbar, Douglas	dldunbar@prodigy.net
Dunnet, Ron	ron@qubbesoft.freemove.co.uk
England, William	wengland@iname.com
Feng, Al	alfeng@juno.com
Fink, Mike	domino.cubes@excelsior.net
Fink, Mike	domino.cubes@pointblank.com
Firshman, Tony	tony@firshman.demon.co.uk
Franke, John	j.m.franke@larc.nasa.gov
Ganger, Gary	gangerg@dma.org
Gilbert, Robert	weena@netzero.net
Gilbert, Robert	mcs_rlg@hotmail.com
Gillespie, Doug	aa431@cleveland.freenet.edu
Girnius, William	girnius_w@bls.gov
Gowen, Rod	ej377@freenet.carleton.ca

Harbit, Ken	krh03@cvip.fresno.com
Henderlight, Mike	mikehend@microsoft.com
Henn, Fred	oranur@juno.com
Humphreys, Rod	rodh@lightspeed.bc.ca
Hunkins, James	jdhunki@ibm.net
Impellizzeri, John	jimpellizzeri@compuserve.com
Jaap, Matthias	matthias.Jaap@hhs.hh.schule.de
Jonas, Mike	mjonas@bbn.com
Jones, Dilwyn	dilwyn.jones@dj.softnet.co.uk
Jones, Terry	tjones@iname.com
Kaczor, Jon	75363.1127@compuserve.com
Kahale, Abed	akahale@compuserve.com
Kealy, Harriet Joan	hikealy@admin.hilconet.com
Kenny, Larry	larken@storm.ca
Kingsley, Ed	edk4@aol.com
Kondrac, Mike	mkandrac@aol.com
König, Urs	urs.koenig@agrodata.ch
Kwitkowski, Phillip	kwit47@aol.com
Lancaster, Garry	dharkhig@delphi.com
Lanciault, Francois	francois.lanciault@energies.alstom.ca
Lassov, David	emanon@azstarnet.com
LaVerne, Melvin	mlaverne@usit.net
Lebowitz, Dave	dki@dpliv.com
Liebert-Adelt, Peter	p.liebert@t-online.de
Malloy, Bob	74776.2342@compuserve.com
McBrine, William	wmcbrine@clark.net
McKelvey, William	mckelveyw@delphi.com
Merz, Jochen	jmerz@t-online.de
Merz, Jochen	jochenmerz@j-m-s.com
Miller, Seymour	seymil@delphi.com
Muth, Bob	bobkeeper1@aol.com
Norton, Gary	gnorton@world.std.com
Oliger, John	joliger@mindspring.com
Parrish, Gil	gil.parrish@abanet.org
Pashtoon, Nazir	nazir.pashtoon@ingram.micro.com
Paul Holmgren	paulholm@indy.net
Payne, Josh	joshpayne@bigfoot.com
Pazmino, John	john.pazmino@moondog.com
Perry, Luke	doidy34@yahoo.com
Perry, Russ Jr	slapdash@enteract.com
Rampolla, Joe	jprampolla@blazenet.net
Rigter, Wilf	rigter@cafe.net
Rish John	74601.1535@compuserve.com
Shepard, Jay	jshepard@netins.net
Simon, Thomas	73177.333@compuserve.com
Skapinski, Tom	tskapins@juno.com
Smith, Dennis	denny.smith@juno.com
Solly, David	ac355@freenet.carleton.ca
Stegman, Dan	danesteg@juno.com
Swenson, Tim	swensontc@geocities.com
Swenson, Tim	swensont@sirclive.csd.sgi.com
Swentko, Wally	wswentko@maroon.tc.umn.edu
Swoger, Robert	ceng108@email.mot.com
Taylor, Jeff	jetaylor@mdrobotics.ca
Thoresen, Jeff	74200.257@compuserve.com
Waldman, Stephen	brogine@hotmail.com
Walterman, Don	walterm@ix.netcom.com
Watson, Keith	keith_watson@juno.com
Wood, Roy	qbranch@qbranch.demon.co.uk
Zimmerman, George	gzimmer928@aol.com



# From The Chairman's Disk

Donald S. Lambert

I have started a project that I will describe. I am going to try to marry a Weirdo to a TWAIN. A few definitions are in order. A Weirdo is anything plugged into the Z88's 9 pin serial port according to the book "Z88 MAGIC". A TWAIN is "Technology Without An Important Name" and by definition a standard for acquiring graphics, such as input from a scanner, computer video, or something else that brings graphics into the computer. And text is a form of graphics.

Here is a kit (Z88 PC LINK II) that contains the hardware and software to interchange Lotus and WordStar files between your Z88 computer and an IBM PC compatible computer. It comes in a box that is 7.562 X 7.562 X 2.312 inches. The box contains a 5.25 diskette, ROM card for the Z88 and a Z88 serial cable with a 25 way D female connector and a PC-LINK USER GUIDE. The box also contained a 3.5 diskette with a hand written label that I worked from

Initially I tried to copy the disk to hard drive by way of DOS and Windows and I kept getting all kinds of reasons why it could not be done with the majority of the reasons stating that I was not in DOS. Finally I read the User's Guide. There I found the instructions for installing PC-LINK to either another floppy or to the hard drive. So at first I tried to get it on hard drive. Step one was to Move onto the hard drive by typing C: Aha! From having looked at my DOS 5 User's Handbook I knew that was a DOS command. Next was to type cd \ to move to the root directory. Create a suitable directory for the PC-LINK software by typing: md PCLINK. Move into the new directory by typing: cd PCLINK. Copy the PC-LINK software onto the hard drive by typing: copy A:\*.\* You will see the files as they are copied across.

But after that I read further in the manual and learned that when files were transferred from the Z88 to the PC they are put into whatever memory the software was in. And to avoid overloading the hard drive (is that possible with a 13gig hard drive?) I decided to transfer the files to a three and a half inch 1.44 meg floppy. I also copied it to a 720K 3.5 floppy. But for some reason the 1.44 meg floppy was listed as a 720K.

Then I discovered that I had to copy the MS-DOS file COMMAND.COM to be with the PC-LINK files be it on hard disk or floppy. So every time I tried that the PC objected and stated that I was either in DOS and couldn't do it or that I was in Windows and couldn't access DOS files. After having the computer lock up a couple of times I decided to copy the files I had on hard drive in case the computer crashed. That is the word-processor files I had created. In doing that I had to read up on the procedure of how to do it. And of course I ran across the procedure for making an emergency back up disk for use if I had to reconstruct the hard drive programs. I had already done that and so I checked that disk and I found the file COMMAND there in the directory.

The manual for PC-LINK has the procedure for copying COMMAND to the disk. It is not

complicated but to copy that one file I had to change disks since I only have one 3.5 drive. I had to put the original and the copy in several times (I think a total of 4 each) before the process was finished.

Now with that hurdle over with I had to connect up the Z88 to the Compaq. The cable has a male nine pin D to connect to the Z88 9 pin female. That is fine and dandy. But the other end of the cable is a 25 way female and the Compaq has a 9 way male. So I needed a serial adapter for the cable. So I had to go to Staples to get one.

I also have a LASER PC3 that is somewhat like a cheap version of the Z88 which also can be up and down loaded to a IBM compatible PC. Somehow I lost the disk of software that came with the LASER PC3 to be loaded into the PC to receive the files or to send the files.

## Is there anybody out there that can send me a copy on a 3.5 floppy? Or even on a 5.25 disk?

Plugged together with all powered down I then powered up and tried the file transfer from the Z88 to the PC. The window of the Z88 directory came up on the PC and I proceeded per instructions but! I never did see the file in the PC. That is where I am stuck right now. Is it lost in cyberspace or is it hiding some place in the PC where I haven't thought to look. I even looked in the directory of the 3.5 disk and didn't see anything

Hopefully, I will get more insight into the problem by the next issue. I will be reading and trying this and that till it works. And when it works I will be reporting it

```
C:\>DIR A:\
```

```
Volume in drive A is EMPTY
```

```
Volume Serial Number is 374D-3BBF
```

```
Directory of A:\
```

ISOIBM	EXE	12,271	08-01-97	3:57a
IBMISO	EXE	12,271	08-01-97	3:58a
PCLINK	EXE	97,277	08-01-97	3:58a
READ	ME	2,762	08-01-97	3:59a
PCLINK	HLP	7,369	08-01-97	3:59a
SETUP	BAT	5,818	08-01-97	3:59a
WTP	EXE	32,471	08-01-97	3:59a
PTW	EXE	43,597	08-01-97	4:00a
PTL	EXE	64,164	08-01-97	4:01a
LTP	EXE	64,364	08-01-97	4:01a
		10 file(s)	342,364 bytes	
		0 dir(s)	385,024 bytes free	

Since you are in DOS, I would make a new directory, then copy from A: to that directory.

```
C:\>mkdir PCLINK
```

(Make dir.)

```
C:\>A:
```

```
A:\>copy *.* C:\PCLINK
```

```
C:\>DIR > PRN
```

(To print dir.)

# FUN WITH HARDWARE PART I - THE PLATFORM

By Jeff Burrell

## INTRODUCTION

**M** first computer was a Sinclair ZX-81 that was purchased while I was in the U.S. Air Force stationed in the U.K. At that time I was a technician working on advanced microwave satellite communications equipment and had access to a Hewlett-Packard minicomputer with a FORTRAN compiler. This, along with my programming experience in college, led me to purchase the ZX-81 when I saw it advertised in a British electronics magazine. Over the five or so years that I used that computer, I learned to program in BASIC and hand-assembled machine code and to use the features and limitations of the "little black door-stop" to the fullest. While stationed at Offutt AFB, I had a co-worker that was moving up to a CP/M based computer from his TS-2068. When I saw the 2068 I knew that I wanted it. I had been impressed by the functionality that Clive Sinclair had managed to squeeze into the ZX-81 and from examining the documentation of the TS-2068 I saw that this was a machine with much promise. I bought the machine and started programming right away and found, as expected, that there was much to like about the 2068.

**A**s time moved on, so did I, and so the 2068 was packed away in a closet. I moved on to MSDOS based computers because of the needs of my job and forgot about the 2068 for a few years. I did finally unpack it to try some hardware experiments with a Memopak modem and much to my chagrin blew out several address lines on the CPLD, effectively making the machine worthless. By this time Timex was well and truly out of the computer business, the internet was still in its

infancy, and I had many other responsibilities, so I repacked the computer and put it back into the closet mostly for sentimental reasons. My present position is principal design engineer for a medical device company doing primarily analog design. About a year ago, the pace went from insanely busy to simply insane and I was looking for a hardware project to do at home. I was especially looking for something that was primarily a digital design since I had not done anything in the digital domain for several years. I was surfing the net looking at various retro game and computer sites when I ran across several web sites devoted to the Sinclair computers.

**W**hen I found out that there was still an opportunity to work with the 2068, I took it. With help from Abed Kahale, Alvin Albrecht, and especially Jack Boatwright I managed to get my hands on a mostly working 2068. Its only deficiency is that the TV output is inoperative but this is not a problem since I have a composite monitor that works well with the 2068. My surfing led me to Alvin Albrecht and his idea for an FPGA for expanding the capabilities of the 2068. I worked with Alvin for a while and have since lost contact with him (Alvin, if you are out there please e-mail me). His ideas meshed well with my own, and I decided to design and build a prototype that would test some of the ideas he and I discussed.

## SPECIFICATIONS

I developed the following specifications based on a combination of hardware availability, my personal desires for learning new skills, and the fact that the first prototype would be a hand-wired point-to-point affair. See figure 1

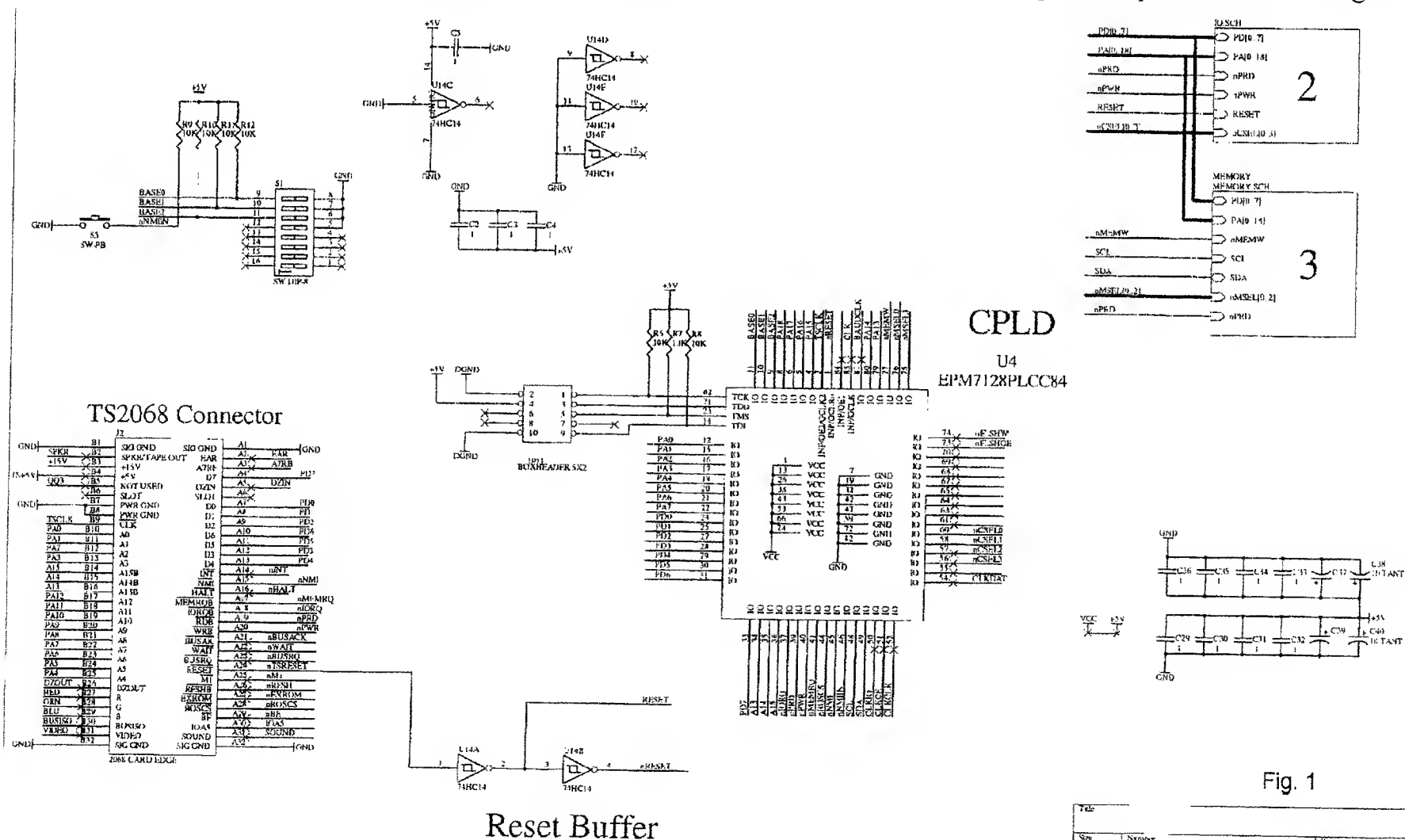


Fig. 1

for the prototype schematic.

**MEMORY** 512K SRAM  
256K Flash  
8K serial EEPROM

**PERIPHERALS** IDE drive interface Centronics  
compatible printer Interface High speed  
synchronous serial Interface using FPGA  
or CPLD Memory is accessed using  
nROSCS from the 2068 bus.

## CONTROL

All I/O ports will be switch selectable in the \$00 to \$7F region to be compatible with the TS-2068 architecture. I had purchased several 512K SRAMS when I upgraded my Z88 and had them on hand. I wanted to allow plenty of elbow room for programs in my expanded 2068 and thought that 32 8K RAM banks would allow for code switching between several resident programs. The 256K flash devices (28F020) were available from PC controllers we use here at the shop (we can't reuse them in product and there are only so many that we can use in prototyping). I think of the flash as primarily a solid state disk that is also available as memory at boot time through the normal dock bank services to initialize and install any extensions to the 2068 OS. I have also thought that this could be used to completely replace the onboard 2068 OS to fix the ROM bugs listed in the technical manual as well as allow the channel services to be expanded past those provided in the original OS release.

The one problem with these devices is that the entire 256K array must be programmed at one go. This makes them rather cumbersome to use as normal non-

volatile memory and more suitable as a flash disk. Digikey stocks a 4Mbyte flash module made by Toshiba that cost about \$17.00 and are configured to act like a flash disk with nominal 512 byte "sectors." These devices look very interesting and I may incorporate them if I can "dead bug" one onto the prototype. For you Z88 folks (like me) add a PIC controller for a serial link and command processor and these would make really good mass storage for you too! Ah, so much hardware, so little time.

The 8K serial EEPROM was used in a previous product and would make a good place to store configuration information and perhaps development code before burning it into the flash.

I have always been chafing at the fact that the 2068 did not have good mass storage available. I did purchase an Exetron Stringy Floppy for my ZX-81 and found it to be eminently useful, but still not up to par with even floppies. I had also thought about designing an IDE interface, so this gave me the chance. The web has several very good resources on the IDE specification and I used them to help design the interface. This implementation uses an 8255 PIA to drive all of the lines to the drive. This will be slower than if the drive was connected directly to the bus, but I thought that since the file sizes are relatively small, the perceived performance should be acceptable. The pull-up resistors on the IDE 8255 are used to ensure that the drive does not get accessed on power-up when the 8255 lines are all inputs and the levels are floating. By pulling the device selects (nDDCS0, nDDCS1) and the read and write controls (nDDRD, nDDWR) high, the drive will not be inadvertently selected.

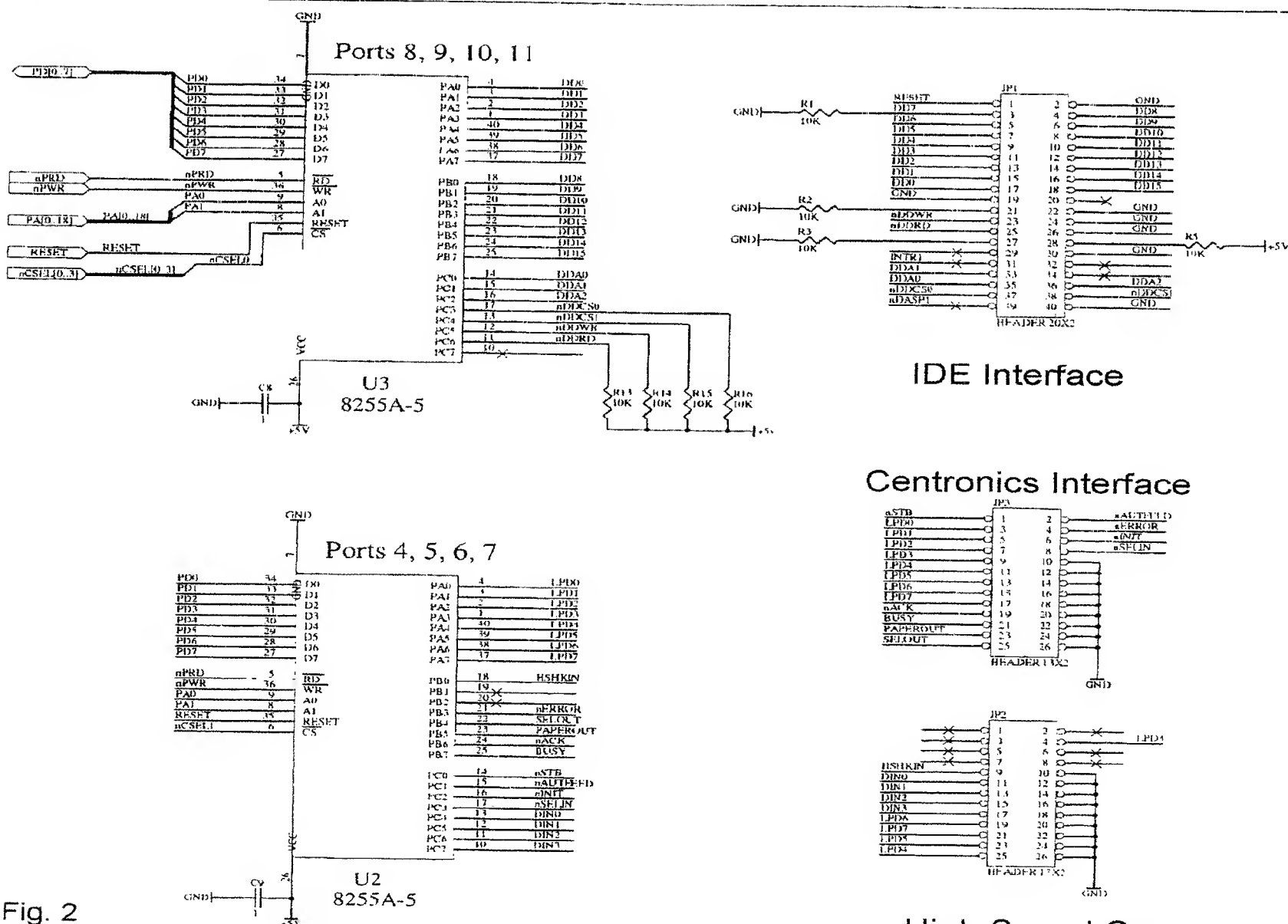


Fig. 2



It seemed to me that I had to include a Centronics port if at all possible simply because it is so useful. Implementing the port with another 8255 gives me the flexibility to make it whatever I want. At the very least it allows me to use a printer with the 2068 and it would also allow use of ZIP or LS120 drives with the machine.

**T**he 8255 Centronics chip has several unused I/O lines that are used for a dedicated high-speed link that allows communication with a PC through the PC parallel port. I thought that using a semi-parallel port for high-speed communication (ala Zip drives etc.) would allow for fast uploading and downloading to and from the 2068 to my PCs. This would allow for developing code on the 2068 or the PC and transferring files either way. My PCs are on a network here at home and it would be nice to have the 2068 access the network through this high-speed link.

The control logic for the board can be complex, so I decided to use an FPGA (Field Programmable Gate Array) or CPLD (Complex Programmable Logic Device) for the control logic. I settled on the Altera EPM7128SPLCC84 primarily because

1. I thought that it would have enough logic resources
2. The device is flash based, so it is programmed on power-up. SRAM based FPGAs require several milliseconds to configure from their external serial EEPROMs.
3. I had access to the Altera development tools
4. The devices were available at work. My boss kindly agreed that I could use the experience.
5. The CPLD is in-circuit programmable - no special programming hardware or software is needed.

**A**s it turns out, the 7128 is a little small for this project. The memory mapper circuit requires 64 latches, and the other functions require somewhat less than that. During the design process I found that I could not fit all of the desired functionality into the 7128. It turns out that it was a routing problem and not a deficiency in logic resources. I did find that I could fit 4 map registers into the chip (16K banks) so I opted for that solution. This is a little disappointing, but I thought that it would still make a useful project. I will eventually turn the prototype into a PCB and, when I do, I will either use a 7160 CPLD or an EPF10K10 FPGA. Either of these devices will give me the extra logic to implement a full memory mapping scheme. The CPLD also performs the I/O address decoding and provides the port to access the serial EEPROM.

**T**he prototype was hand-wired on a Vector pad-per-hole perf-board. I chose point-to-point wiring primarily because it would allow the prototype to fit onto the back of the 2068; wire wrap pins would have been much too long. I also have had good luck with point-to-point wiring on analog and digital prototypes of this complexity in the past. I will try to post a photograph of the prototype in the future.

### DEVELOPMENT TOOLS

My software development tools are a rather mixed bag. I am using a cross assembler that has no relocation capability. This has not been a problem so far, but I am looking towards using an assembler I found on the **Z88 Forever** website. Even though I usually program in

BASIC, I have also found a C compiler on the same web site that I will also evaluate for coding most of the rest of the operating system code. I have heard mention of a compiler for 2068 BASIC and think that it could make an interesting alternative.

**I** found that I needed a way to step the code to test it out before crashing it on the 2068. I modified a public domain Z80 emulator that I found on the net to include emulation of my interface. This required translating the emulator to my Power BASIC compiler. I have used the DOS and Windows versions of Power BASIC for about ten years and found the language to be easier to use than C and very fast - it is a true compiler not in interpreter like Visual BASIC. I highly recommend this language for use on the PC where portability to other platforms is not an issue. I wrapped a Windows interface (again using Power BASIC) around the emulator and have been testing it for the last few days. There are still a few bugs, but it seems to be nearly finished. I have been using it to step through my memory management code and have found it to be invaluable even in its present state.

**I** also wanted the source code for the 2068 ROMs to use as a guide as well as a basis for modifying the OS. I modified my emulator's disassembler to allow the use of a table that contained the addresses of the various modules in the Home ROM and EXROM. The table was generated using data from the PDF version of the 2068 technical manual that Alvin Albrect put on the web. The use of the tables allowed generating a preliminary disassembly of the ROM images I found on the net. I then massaged the listing to make it compatible with my assembler and started adding comments by using Ian Sinclair's complete ZX81 disassembly. This was most useful in decoding the floating-point routines along with some of the tables and data areas used by the interpreter. I also used a listing of the Spectrum ROM (comments in German) along with hints from the 2068 technical manual to generate a final disassembly that had all of the data tables properly formatted. Finally, after all of these gyrations, I was able to assemble the final products and obtain binary images that exactly matched the ROM images - hooray! I still don't have a completely commented disassembly and I have not sussed out all of the operations in the OS, but that is only a matter of time.

### GENERAL TESTING

I have so far tested the operation of the memory mapper and memory control, and the 8255 interface. I ran into an interesting bug when testing the memory mapping circuitry. When I designed the memory mapper, I used the nROSCS to qualify the control signals to the memories assuming that it was only active during memory accesses. Everything seemed to be working well in the initial testing and so I started more in depth testing. During the more detailed testing, I found that when I wrote to the mapper registers, the data was also written to the added RAM. I discovered that the nROSCS was active not only during memory accesses, but also during I/O accesses if the high bits of the address bus were in the proper state. This was caused because the memory mapper registers use all 16 bits of the address bus (via the OUT (C), A and IN A, (C) instructions) to decode their addresses to simplify the mapper selection logic. The upper two bits of the B

register select one of the four mapper registers for reading and writing (at I/O addresses \$0000, \$4000, \$8000, and \$C000). The low eight bits of the I/O address are used, as normal, to specify the I/O block to be accessed. Because of the way the 2068 SCLD works, the nROSCS is brought low based only on the address lines and is not qualified with nMEMSEL from the Z80 – EXACTLY as described in the documentation. Because I assumed that nROSCS was active only during memory accesses, I was writing to both the RAM and the desired mapper register. After qualifying the RAM control lines with the processor nMEMSEL I obtained the desired operation. RTFM.

## HS COMMUNICATIONS

Testing of the “high speed” serial link has progressed well. So far I have just verified that the link is operational. I wrote several small programs to transfer bytes in both directions between my PC and the 2068. I then wrote the 2068 program in Listing 1 to upload itself via the serial link to the PC. The listing is a translated version of the byte stream sent to the PC. This approximately 1K of code took nearly 30 seconds to transfer. I am now in the process of writing an assembly program for the 2068 to speed up the process. I do sorely miss bit oriented functions (AND, OR, XOR, and shifts) in the 2068 BASIC dialect as this would eliminate many of the multiplies and divides in the program.

### LISTING 1:

```
0010 GOSUB 9000
0015 GOTO 100
0020 GOTO 200
100 LET x=PEEK 23635+256*PEEK 23636
0110 FOR i=x TO x+1024
0120 LET b=PEEK i
0130 GOSUB 4810
0140 NEXT i
0150 STOP
0200 FOR i=1 TO 65
0210 GOSUB 4900
0220 PRINT "received: ";b
0230 NEXT i
3999 STOP
4800 REM send a byte
4810 LET n=16*INT (b/16)
4820 GOSUB 5000
4830 LET n=16*(b-n)
4840 GOSUB 5000
4850 RETURN
4900 REM receive a byte
4910 GOSUB 5200
4920 LET b=16*INT (n/16)
4930 GOSUB 5200
4940 LET b=b+INT (n/16)
4950 RETURN
5000 REM send a nybble
5010 LET h=IN 5
5020 IF 2*INT (h/2)=h THEN GOTO 5010
5030 OUT 4,n+8
5040 OUT 4,n
5050 LET h=IN 5
5060 IF 2*INT (h/2)<> h THEN
GOTO 5050
5070 OUT 4,n+8
5080 LET h=IN 5
5090 IF 2*INT (h/2)=h THEN GOTO
5080
```

```
5100 RETURN
5200 REM receive a nybble
5210 LET h=IN 5
5220 IF 2*INT (h/2)<> h THEN GOTO 5210
5230 LET n=IN 6
5240 OUT 4,0
5250 LET h=IN 5
5260 IF 2*INT (h/2)=h THEN GOTO 5250
5270 OUT 4,8
5280 RETURN
9000 REM set up 8255
9010 OUT 7,138
9020 OUT 4,255
9030 RETURN
```

## MEMORY MANAGEMENT

I have started writing a memory manager loosely patterned after the LIM 4.0 standard used in PCs. I chose this route because the LIM 4.0 standard makes a good framework for both managing memory for data storage and running overlays in the expanded memory area. It would also allow a multitasking kernel to be developed that would allow some very interesting playing with the 2068. At this point I am still finishing the routines and only have the memory movement subroutines to write. These routines will be rather interesting because I want them to be flexible enough to move data from anywhere in the home, dock, or EXROM banks to anywhere else in those banks. The trick comes because the memory move code must execute from somewhere and that somewhere may be in one of the memory spaces to be specified in the move command. I have examined the function dispatcher code to see how Timex did it and have some ideas to try, but this will be a messy bit of code none-the-less. Since memory operations are fundamental to any computer, once these routines are written I can move on with the other stuff

## LAST ITEM

I had mentioned to Abed the possibility of making two more of these interfaces for loan to others for software and hardware development. I would be interested in hearing from anyone who may be seriously interested. It will take about two months to build the interfaces because I will want to lay out a PCB to aid in construction and make a more predictable product.

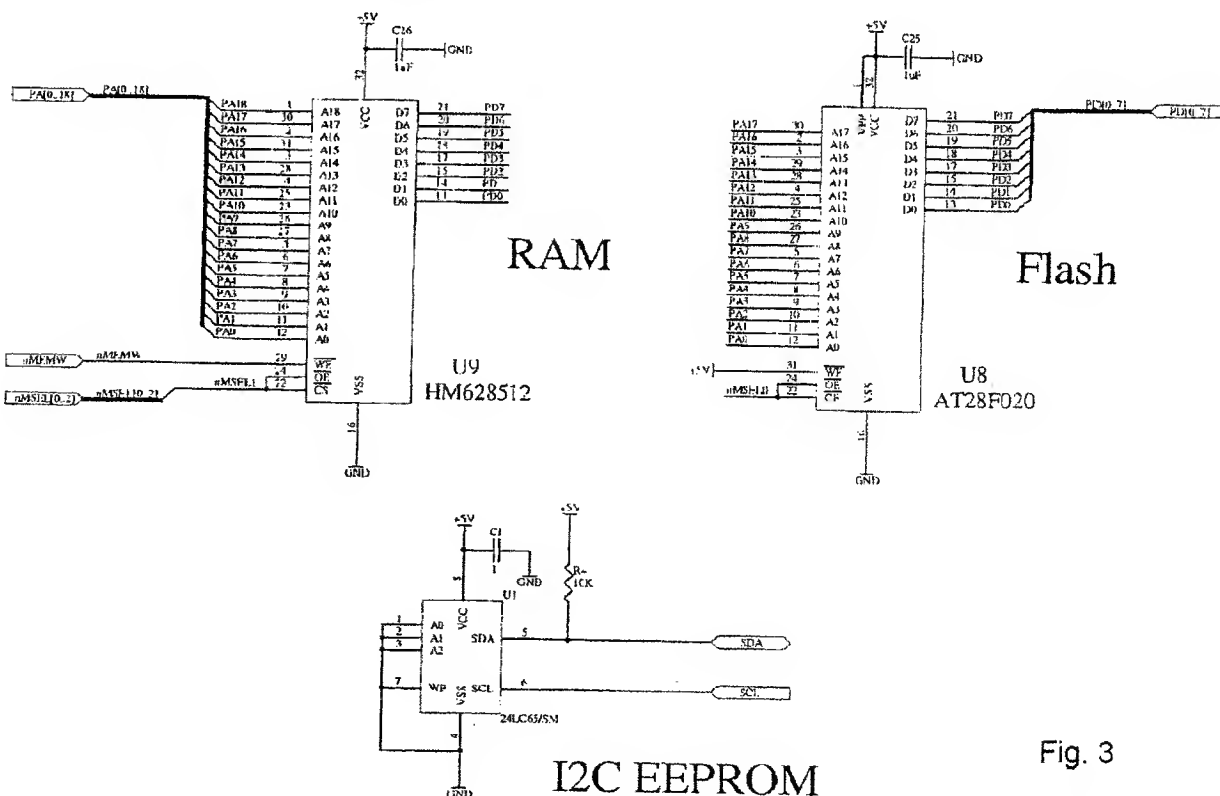


Fig. 3

# QL Hacker's Journal

Supporting All QL Programmers

#32 December 1999

The QL Hacker's Journal (QHJ) is published by Tim Swenson as a service to the QL Community. The QHJ is freely distributable. Past issues are available on disk, via e-mail, or via the Anon-FTP server, garbo.uwasa.fi. The QHJ is always on the look out for article submissions.

QL Hacker's Journal

c/o Tim Swenson

2455 Medallion Dr.

Union City, CA 94587

swenson@lanset.com

<http://www.geocities.com/SiliconValley/Pines/5865/index.html>

## Editor's Forum

**F**irst off, thanks to Herb Schaaf for providing most of the text for this issue. When Herb gets a hold of a problem, he does not let go until he knows all there is to know.

Secondly, I wanted to bring up something that has been discussed in the QL-users mailing list. I can't remember who started it, but there was a discussion on what software QLers would like to see next for the QL. The obvious, and non-trivial, stuff was mentioned; a TCP/IP stack, a HTML browser, color graphics and software to use it, and so on. As much as these are nice, the level of effort to create them is beyond most QL programmers. What I would like to see is a list of more day-to-day applications with a much lower level of effort. Once a list is created, QL programmers could see what others want and take a try at writing the programs. As much as I write software for my own needs, it's nice to know when others also find it useful.

**T**o help this idea along, I'm willing to act as the go between the user and the programmers. I'll take program suggestions from users and list them on my web page. From there, programmers can volunteer to work on a program, which I can coordinate so that two people are not working on the same program. So, if you have some ideas of what software you would find useful, send it to me.

## Befunge

**I** recently found a new language that has an implementation that will run on the QL - Befunge. Befunge is the first two-dimensional language. This means that the Program Counter, instead of just going down a line of code, can move up, down, left, and right. This is easily done by making each Befunge command a single character.

Like FORTH and Postscript, Befunge is a stack oriented language. Each command either is data to be put on the stack, or is an operator to do something with what ever is on the top of the stack. Numbers are limited to input via single digits, but may be expanded by manipulating the stack. To get 23, you would push 7 on the stack, push 3 on the stack, do a multiply command (leaving 21 on the

stack), push 2 on the stack, and do an add command (leaving 23 on the stack). In Befunge it would look something like this: 73\*2+

To then print out the value on the stack, just use the integer print command, the period (.).

**B**efunge code space is an 80x24 array of characters. The Program Counter starts off in the upper left hand corner and moves from left to right. It continues this way until it hits the end of a line and wraps around to the start of the same line or until it hits a command that changes its direction.

**I**would love to talk about how fun and easy the language is, but I still have not quite figured it out. The examples used in the Befunge language documentation look easy, but the example code that comes with it is still causing my brain to hurt.

**N**ow as far as running Befunge on the QL - let me start with a little history on how I ran across Befunge. The Fall 1999 edition of "The Perl Journal" had the results of the latest Obfuscated Perl Contest. One of the entries was a Befunge interpreter. Since the code was rather cryptic (as is all Obfuscated programs) and I did not know if there were any Perl5-isms in the code. I tried to get a copy from "The Perl Journal" web page. Having forgot my subscribers account name and password, that did not go well. Time to try Yahoo! and do a search on Befunge. Look, a nice Befunge web page, with documentation and a list of implementations. One of the implementations is in C. Not being a C person, I was not up to the tasking of porting it to the QL. Then I saw the answer, an implementation in Z-code.

**F**or those that don't know, Z-code is a data file used for the old Infocom adventure games. Infocom adventure games are written in the language Inform, then compiled, and then run with ZIP (Z-code Interpreter). A version of ZIP has been ported to the QL and I've used it a number of times over the years. I downloaded the Z-code implementation of Befunge, unzipped it and fired off ZIP with zbefunge.Z5 as the data file. After some initial complaining about not been run on a "real" Z-code interpreter, I was able to the main screen up. The Z-Code version has a built in editor, from which you can then run a program.

**I**typed in the "Hello World" example program, hit the function key to run the code and "Hello World" was printed out on the screen. It looks like it works.

Once nice feature of the Z-code implementation of Befunge, is a debug option. Using this option, you can step through the Befunge program watching the program counter move about. As the program runs, you can see, at the bottom of the screen, the top 6-or-so values on the



stack. You can also toggle over to a screen where the output is shown.

Using the debug feature I am starting to get a hang on how the Program Counter moves about.

**T**his implementation also allows for loading and saving Befunge programs. I tried loading some of the programs that come with Befunge, but the interpreter had an error. I then typed in the "Hello World" program and saved it. When I loaded it back in, I got an error, but the program then showed up fine. It looked at the saved file and it has a bunch of extra non-ASCII stuff. It looks like the saving and loading feature has some problems, but it does seem to work.

The main web page for Befunge is:

<http://www.cats-eye.com/befunge/>

The web page to get the Z-code implementation of Befunge is:

<http://www.meta.demon.co.uk/zbefunge.html>

Here is a few example Befunge programs:

Listing 1: (Hello World)

```
v
>v"Hello world!"0<
,: ^_25*,@
```

Listing 2: (Factorial)

```
v
>v"Please enter a number (1-16) :
"0<
,: >$*99g1-:99p#v_.25*,@
^_&:1-99p>:1-:!!10
^ <
```

Listing 3: (To Upper)

```
v, < < <
>~:"a"1-`!#^_:"z"1+`#^_"aA"--^
Perlpull prose, (required reading)
by Herb Schaaf
```

"a perl of great precise(sic)", but limited accuracy.

"Optimized for text" they say of perl, that wonderful Swiss-army chainsaw programming language. But I've had great fun with numbers, discovering the mathematical abilities of perl for the QL, version 4.036 as ported over by Jonathan Hudson. We get answers to math problems in double precision, carried out to 14 or 15 significant figures, similar to ABACUS. When I tried Tim's dice.pl (QHJ#30), and put in 2 die with 3 sides, or 3 die with 2 sides (like flipping coins), I got very strange answers. Things (\$totper) didn't add up to 100%, but would come out under or over. I discussed this with Bill Cable while at the East Coast QL show. He got good and reasonable answers on his PC laptop using Perl 5, but with QL perl 4 under QLayer he was able to get the same funny answers as I had found.

**T**urns out to be in the exponentiation function. This creates a floating point number, and these can cause trouble when used for counting or comparisons. Exponentiation uses natural logarithms and rounding errors in the 15th decimal place cause the bogus answers. I wrote a perl subroutine using an algorithm similar to the one in the power function from "The C Programming Language" K&R 2nd ed., section 18, page 27 which uses integer values and so far has given the "right" answers. Perl does not use typecasting, so we can't

declare (int)power. Nor can I try "use integer," to see how that works.

**I**n the dice.pl program I changed from the exponentiation operation to a call to the power subroutine and then the answers came out as expected.

**B**ut you might ask; how can we make a call to a function or procedure in perl? The answer is the use of the perl keyword "sub" before the name of the function block which is appended to the listing, and the use of the ampersand "&" before the name of the function to call it. Another choice is to write the subroutine as a file which you add to your library and can then pull it in when wanted by simply asking your main program to "require" it.

Perlpull prose - my adventures with perl pulls prose out of me.

Purple prose - expletives replete when recursing a perl problem.

Here's the block to append to dice.pl:

```
sub power {
    local($base,$exponent) = @_ ;
    $power = 1 - ($base == 0);
    if (($base == 0) && ($exponent ==
0)) {
        $power = "NaN";
    }
    else {
        while( $exponent > 0) {
            $power *= $base;
            $exponent--;
        }
    }
    return $power;
}
```

To put it into the library, think of a filename for it; (I used lib\_power.pl) and add:

1;

as a final last line.

In QLHJ#30 the three instances of the expression:

(\$sides\*\*\$num\_die)

are replaced with the expression:

(&power(\$sides,\$num\_die))

**N**ow you can either add the power subroutine (without the final 1;) to the dice.pl listing, or you can pull in the library version (with the final 1,) by having a line added at the beginning of dice.pl that reads:

```
require "power.pl";
```

which will pull it in from your lib\_subdirectory.

To see how the values compare, try this compower.pl program:

```
#!/usr/bin/perl
# compower.pl for QL perl 4
# H L Schaaf August 21, 1999
# to compare the results of exponentiation in perl with
# a method that multiplies an integral number of times.
$around = 1;
$log_limit = log(2**1023);
while($around){
    print "\fThis is round ", $around;
```

```

    print "\n\n\t please ENTER a number
for the base ";
    $base = &inkey.<STDIN>;
    chop $base;
    if($base) { $safe_size =
&abs(int($log_limit/(log(&abs($base))))
);
    }
    else {
        $safe_size = 2**1023;
    }
    print "\n\t(exponents larger than
",$safe_size," are
    probably too large)";
    print "\n\n\t please ENTER a
positive integral number
    for the exponent ";
    $exponent = &inkey.<STDIN>;
    chop $exponent;
    $power = &power($base,$exponent);
    print "\n\n\t          integral power
",$power;
    $float_power = $base**$exponent;
    print "\n\n\t floating point power
",$float_power;
    print "\n\t -----
---";
    print "\n\n\t          difference is
",$
$power-$float_power;
    print "\n\n\t ENTER for another,
ESC (at any time) to
quit";
    &inkey;
}
continue {
    $around ++ ;
}
sub power {
    local($base,$exponent) = @_;
    $power = 1 - ($base == 0);
    if (($base == 0) && ($exponent ==
0)) {
        $power = "NaN";
    }
    else {
        while( $exponent > 0) {
            $power *= $base;
            $exponent--;
        }
    }
    return $power;
}
sub inkey {
    sysread(STDIN,$inkey,1);
    if (ord($inkey) == 27) {
        print "\b \n\n\t\t";
        exit;
    }
    return $inkey;
}

```

```

sub sgn {
    local($n) = @_;
    return ($n <=> 0);
}
sub abs {
    local($n) = @_;
    return ($n * &sgn($n));
}

```



Other folks have written all sorts of things for perl that can be "required" and used. Two numerically interesting examples, bigint.pl and bigfloat.pl (which itself pulls in bigint.pl), are in the library provided by Jonathan Hudson. I found them fun to noodle with, so why not give them a try if you're into math and want to see results carried out with great precision. You can set the number of significant digits to be "arbitrarily (?)" large.

Here is the result of my noodling around:

```

#!/usr/bin/perl
# bigfloatdemo.pl bigfloat.pl and
# bigint.pl in QL perl 4
# H L Schaaf August 21, 1999
print " a small demo of big floating
point and big integer operations in
perl";
print "\n please wait for required
module(s) to be pulled in from the
library";
require "bigfloat.pl";
$around = 1;
while($around){
    print "\f this is round ", $around;
    print "\n please ENTER the first
number ";
    $n1 = &inkey.<STDIN>;
    chop $n1;
    $valid_answer = 0;
    print "\n choose an operation by
touching the appropriate key\n";
    print "\n      [P]lus,      [M]inus,
[T]imes, [D]ivided by ";
    print "\n      [R]aise to an integral
power, [S]quare root (these take
time)";
    print "\n\t\t\t [G]reatest common
denominator\t";
    while( !$valid_answer){
        $op = &inkey;
        if ($op =~ /[pPmMtTdDrRsSgG]/)
        {$valid_answer = 1;}
        print "\b \b";
    }
    print "\n\n";
    if($op =~ /[sSdD]/) {
        print "\n How many signifigant
digits wanted ? ";
        $sig_digits = &inkey.<STDIN>;
        chop $sig_digits;
    }
    else {
        $sig_digits = 1;
    }
    if ($op =~ /[sS]/) {

```

```


        print "\n\t\t please wait \n";
        $started = time;
        $f = &fsqrt($n1,$sig_digits);
    }
    else {
        print "\n please ENTER the
second number ";
        $n2 = &inkey.<STDIN>;
        chop $n2;
        if ($op =~ /[rR]/) {print "\n\t\t
please wait \n"; }
        if ($op =~ /[pP]/) {$f =
&fadd($n1,$n2,$sig_digits); }
        if ($op =~ /[mM]/) {$f =
&fsub($n1,$n2,$sig_digits); }
        if ($op =~ /[tT]/) {$f =
&fmul($n1,$n2,$sig_digits); }
        if ($op =~ /[dD]/) {$f =
&fdiv($n1,$n2,$sig_digits); }
        if ($op =~ /[gG]/) {$f =
&bgcd($n1,$n2); }
        if ($op =~ /[rR]/) {$started =
time; $f =
&bpow($n1,$n2); }
    }
    if ($op =~ /[rRsS]/) {
        $elapsed_time = time -
$started;
        print "\n\t\t that took about
",$elapsed_time,"
second", ((($elapsed_time == 1) ? " " :
"s"), "\n");
    }
    print "\n",$f,"\n";
    print "\n ",&withdecimal($f);
    print "\n touch ENTER for another
demo or ESC (at any
time) to exit ";
    &inkey;
    print "\f";
}
continue {
    $around ++ ;
}
sub inkey {
    sysread(STDIN,$inkey,1);
    if (ord($inkey) == 27){
        print "\b \n\n\n\t\t";
        exit;
    }
    return $inkey;
}
sub withdecimal {
    local($bigfloat) = @_ ;
    local($number,$exponent) =
split('E',$bigfloat);
    $decimal_place = (length($number))
+ $exponent;
    if ($exponent > 0) {
        $number = $number."0" x
($exponent);
    }
    if ($decimal_place>1){


```

```


        $bigfloat_with_decimal =
substr($number,0,$decimal_place)
.".".substr($number,$decimal_place);
    }
    else {
        $number_lead
=&substr($number,0,1)."0.";
        $zeros = "0" x (1 -
$decimal_place);
        $bigfloat_with_decimal =
$number_lead.$zeros.substr($number,1);
    }
    return $bigfloat_with_decimal;
}
sub bpow {
    local($bbase,$bexponent) = @_;
    $bpower = 1 - ($bbase == 0);
    if(($bexponent == 0) && ($bbase ==
0)) {
        $bpower = "NaN";
    }
    else {
        while ($bexponent) {
            $bpower =
&fmul($bbase,$bpower);
            $bexponent--;
        }
    }
    return &fnorm($bpower);
}

```

 imagine how we could embellish this by adding trig and other math functions. We could even create a general purpose ("general perlplus"?) scientific calculator with store, recall, memory registers and such; maybe it has been done already and is on CPAN. Of course half the fun is in writing a program yourself, and the other half is debugging and getting it to work.

 tend to think in in BASIC, my first programming language, so have tried to find ways to get results in perl that are comparable to some of the S\*BASIC commands. Here are some equivalents that seem to work:

INKEY\$(-1)

 like to have an interactive menu sometimes, and just want the same action as we enjoy with INKEY\$(-1) in S\*BASIC. I finally found a way to do it on the QL in perl. This gets a key from the user without them having to touch ENTER. The perl keyword is sysread.

inkey.pl is an example that will detect the ESC key from the user

# sysread as a way to read inkey without use of ENTER

# inkey.pl

```

while(1){
    sysread(STDIN,$raw,1);
    if ( ord($raw) eq 27 ) {
        print "\b \t";
        exit;
    }
}

```



```

else {
    $ans = $raw.<STDIN>;
}
chop($ans);
print "\n";
print length($ans);
print "\n", $ans, "\n";
if (length($ans)==1) {
    if ($ans =~ /[yY]/) {
        print "\n That's a Yes \n";
    }
    if ($ans =~ /[nN]/) {
        print "\n That's a No \n";
    }
}
}

```

**I** can't edit the first digit of numbers when I use this inkey.pl, maybe there is a work-around like the getch and ungetch in C? When using QTPI as a link to UNIX at the University of Delaware, this inkey subroutine behaves differently; it still works, but seems to be anticipated or read-ahead in the script. There's probably a better way; how would you do it?

```
REPEAT END REPEAT
```

```
$around = 1;
```

```
while ($around) { }
```

**I** n my explorations of perl programs I often want to keep trying different inputs to see how things go, without having perl exit after my first exploration. By putting everything inside in braces after a while(1) I'm able to get the effect that REPEAT, END REPEAT has.

exit; works anytime to break out of the while(\$around), so we can test for some condition (like the ESC key being touched) to end a session.

I use the continue block to keep count of the trips around the while loop.

```
continue {
```

```
    $around ++;
```

```
}
```

**I** believe that nearly the same effect (except for the continue) could be accomplished very simply with the -n or -p switch on the command line. And of course we could use for loops too. TMTOWTDI or tim-toady as they say in perl; "There's More Than One Way To Do It."

```
CLS
```

```
    print "\f";
```

or formfeed, does the trick when we want a "clean slate".

DATE

Example: \$now = time;

the keyword in perl is time.

perl's calendar starts in 1970 instead of the QL's 1961;



I used the time function in bigfloatdemo.pl to see how long it took to extract square roots or raise to integral powers with the bigfloat.pl and bigint.pl libraries. It took 220 seconds to raise 2 to the 1024th power, and 11 seconds to get the square root of 2 to 100 digits, with the Super Gold Card. With a Gold Card it took 681 seconds for the power and 28 seconds for the root. It took 2 seconds for the power and 0 for the root with Perl 5.005\_02 running under sun 4 solaris using my University of Delaware UNIX account via QTPI. It was nice to see the same program work on both QL's and under the UNIX setup.



When running compower.pl the UNIX results were equal with no difference between the exponentiation operator and the power subroutine. So perhaps Perl 5 does something with integer exponentiation that Perl 4 does not.

```
PAUSE(power_cycles)
```

```
    sleep(seconds);
```

perl's sleep is measured in seconds; the QL PAUSE counts the power line cycles. If no parameter is given both will wait forever. perl has alarm(seconds); but I haven't sussed it out yet. How do you regain control in perl if you've done sleep(); ?

CODE("character in quotes") in the QL

is equivalent to ord(\$chr) in perl this returns the ASCII code for a character.

chr(\$ascii) in perl is equivalent to CHR\$(ascii) in the QL and returns the character for the ASCII code.



We could start a module of these equivalents and conversions between QL S\*Basics and perl and put them into our library as well. We might also try to build an associative array %QLBASIC\_perl\_hash following the example of Bill Cable's "English-Spanish converter" and see how they work and learn how to add more terms to the list.



Perl has other ways to pull in snippets, scripts, etc. and I wonder how the keywords 'do', 'eval', and 'use' work. Anybody want to give us some examples? Oh, there's a LABEL: concept in perl that uses 'next', 'last', and 'redo'. Anybody want to show us how those work? How about the termcap.pl items, can we control the cursor, do ASCII graphics? What about bigrat.pl?

What perl features have you found fun or useful?

# DISKETTE MENU PROGRAM FOR THE TIMEX/SINCLAIR-2068

(LarKen DOS Version)

Article & Program By David Solly

**A** LarKen disk drive is wonderful addition to anyone's Timex/Sinclair 2068. It allows you to store many programs per diskette and to access them rapidly. On the down side, searching through many diskettes looking for a certain program by cataloguing each diskette and hunting or, after not using a diskette for a length of

time, trying to remember what program "CCat3.B1" does can be a major headache. One solution to the problem is the *T/S 2068 Disk Menu*.

How Does The *T/S 2068 Disk Menu* Help?

Once you type in the *T/S 2068 Disk Menu*, (henceforth TSDM), and save it to each diskette TSDM:

- Becomes a permanent part of the diskette which cannot peel off and become lost,
- You can give a descriptive name to each program on the diskette and provide the USR address for machine code programs if they are not the same as the start address (See item "B" in the illustration),
- You can add, delete or modify descriptive names and program file names at will, and,
- TSDM will launch BASIC programs and load screens and machine code programs and display the start address and length of machine code programs if supplied.

```

T/S 2068 Disk Menu
Disk Name: Tool Box #3
Program Select
TTSUG Utilities Menu
Screen Copier 64207
Gorilla Screen Expander-Copier
Pixel Fix
Screen Save/Recall
Character Set Analysis
Line List Plus (Line 9900)
Graphics Replacement
Desk Top Publisher
Copy Cat III Tape Copier
Program Delete
Dos 63488
Hit enter to repeat list
QU = Quit; CAT = Catalogue Disk
Select Letter and Enter C

```

#### How Does TSDM Work?

LL. 100-250

This section resets RAM\_TOP – which is necessary if you launch TSDM using LarKen auto-start – and restores the data read pointer to the beginning of the data. Line 110 opens stream number four to the LarKen disk drive and opens stream number three to a line printer – which is usually an Epson compatible printer. Line 210 prints the banner and line 230 is a DATA statement in which to record the diskette title.

LL. 300-399

These are DATA statements in which you enter the descriptive name for each program you wish to load. There is room on the screen to have twelve one-line descriptive names which are lettered from "A" through to "L". (If you have more than twelve entries, then save several copies of TSDM with slightly different file names – e.g. Menu.B2, Menu B3, etc. – and use the current menu to chain to the previous menu or the next menu.) The program letters must be assigned in order and be the last element in the descriptive name. The reason for this will become clear as we proceed. Line 399 is special because it contains the signal "\*\*\*\*" which tells TSDM to stop reading in data. If you modify this line for any reason, be sure not to remove the data element "\*\*\*\*".

LL. 400-410

These lines read in and display the data in lines 300 to 399 in alternating bands of black ink on white paper and black ink on yellow paper and stops when the signal "\*\*\*\*" is read.

LL. 440-470

At this point TSDM switches to cap-lock mode and you are requested to make a selection or type in the command "QU" or "CAT". "QU" stops TSDM. "CAT" executes the LarKen DOS catalogue

command which lists the contents of the diskette to the screen. TSDM will pause indefinitely until a key is hit at which time TSDM runs itself again. TSDM will continue to loop until you enter your selection.

LL. 480-640

If a single letter is selected then TSDM switches off the cap-lock mode, the data read pointer is set to line 500 and the loop beginning at line 610 begins reading data starting from line 500 until it reaches the line containing the data required to load the selected program. If at this point TSDM encounters the stop sign "\*\*\*\*" then the message "Select Again!" is displayed and TSDM will re-run after a short pause.

Lines 500 to 499 work in much the same manner as lines 300-399 except here is where TSDM finds the data required to load your selection.

Each DATA statement must have three elements.

- ❖ a properly formatted LarKen file name,
- ❖ a program start address, and,
- ❖ a program length address.

Note that the program file names must be entered same order as the descriptive names.

For a BASIC program, enter zeros for both the start address element and the program length element as shown below.

DATA "Tmenu B1",0,0

For machine code programs, you can do as for BASIC programs as stated above. This is how you would load a normally saved screen dump, (i.e. SCREEN\$).

If, however, the machine code program must be loaded at an address other than where it was saved from then you have enter the start address and the program length into the DATA statement thus:

DATA "EpmDOS.C1", 63488, 2047

If you do not know the program length put zero as the last element and the program in its entirety will be loaded at the start address provided

LL. 650-8999

Once TSDM has the load data, it uses the unique LarKen extensions to determine if it is to load a BASIC program or a machine code program. If "B" is the penultimate letter in the program file name, the TSDM proceeds to load the file as a BASIC file. The alternative extension begins with "C". If a "C" extension is not found then TSDM displays the error message "File type error in file", displays the file name and stops. If a "C" extension is present, TSDM then looks at the start and length data stored in the variables "I" and "S" to determine which of the three methods will be used to load the program. If the start address and length elements are other than zero, they are displayed before the program is loaded.

LL. 9000-9920

✓ This is the location of the program save routine for TSDM.

✓ Will TSDM Work With A

✓ ZX Spectrum Emulator ROM?

Yes, it will work when LarKen DOS is being run on a T/S 2068 using a ZX Spectrum Emulator ROM, however, you need to replace line 458 with:

458 IF a\$="CAT" THEN CLS : PRINT #4: CAT: PRINT "Hit any key to restart": PAUSE 0: RUN

Once this is done, TSDM will run perfectly.

## What About The ZX Microdrive?

I have successfully used TSDM on a ZX Microdrive. It is a matter of removing line 110, all the references to PRINT #4, and modify the CAT command, (line 458) and all the SAVE & LOAD commands to the appropriate Microdrive commands. If you save all your programs to the stringy floppy using LarKen file naming conventions then TSDM has no trouble locating and running them. The only possible problem is if you have more than one microdrive. Then you have to look into some means of accounting for the drive number – which I will leave for you, dear readers, or as the topic of another article.

David Solly, Ottawa, Canada, January 10, 2000.

## Program Listing

```
1 REM LarKen Disk Menu
2 REM Program by David Solly
3 REM Ottawa, Ontario
4 REM 9 July 1987
5 REM Updated & Annotated 1 January
2000
100 CLEAR 65535: RESTORE: REM Reset
Ram_Top & Restore data
105 REM Open a stream to the LarKen
disk drive and one to the line printer.
106 REM Both commands are LarKen DOS
specific
110 RANDOMIZE USR 100: OPEN #4,"dd":
PRINT #4: OPEN #3,"lp"
120 BORDER 7: INK 0: PAPER 7
210 PRINT ' PAPER 1; INK 7;" T/S 2068
Disk Menu "
220 PRINT "Disk Name: ";
230 DATA "Tool Box #3"
240 READ n$: PRINT n$
250 PRINT ' INVERSE 1;"Program";
INVERSE 0;TAB 26; INVERSE 1;"Select":
PRINT
299 REM Descriptive file name start
here
300 DATA "TTSUG Utilities Menu A"
302 DATA "Screen Copier 64207 B"
304 DATA "Gorilla Screen
Expander-Copier C"
306 DATA "Pixel Fix D"
308 DATA "Screen Save/Recall E"
310 DATA "Character Set Analysis F"
312 DATA "Line List Plus (Line 9900)
G"
314 DATA "Graphics Replacement H"
316 DATA "Desk Top Publisher I"
318 DATA "Copy Cat III Tape Copier J"
320 DATA "Program Delete K"
322 DATA "Dos 63488 L"
399 DATA " ","Hit enter to repeat list
","QU = Quit; CAT = Catalogue Disk
","****": REM Do not touch!
400 READ n$: IF n$<>"****" THEN PRINT
n$( TO LEN n$-1);TAB 31;n$(LEN n$): GO
TO 410 405 GO TO 440
```

```
410 READ n$: IF n$<>"****" THEN PRINT ;
PAPER 6;n$( TO LEN n$-1);TAB 31;n$(LEN
n$): GO TO 400
440 POKE 23658,8: REM Force caps-lock
mode
450 INPUT " Select Letter and Enter
"; LINE a$
455 IF a$="QU" THEN POKE 23658,0: STOP
458 IF a$="CAT" THEN CLS : PRINT #4:
CAT " ",: PRINT "Hit any key to
restart": PAUSE 0: RUN
460 IF a$="" THEN RUN
465 REM *** Select The File Module
470 IF CODE a$<CODE "A" OR CODE
a$>CODE "Z" THEN GO TO 440
480 POKE 23658,0: RESTORE 500: REM
Release the caps-lock mode, reset data
490 REM Real program data starts here
500 DATA "Tmenu.B1",0,0
502 DATA "Copal.C1",64207,1130
504 DATA "Pander.B1",0,0
506 DATA "Pix.B1",0,0
508 DATA "Dump.B1",0,0
510 DATA "ChaA.B1",0,0
512 DATA "Llist+.B1",0,0
514 DATA "Grep.B1",0,0
516 DATA "DeskPL.BD",0,0
518 DATA "CCat3.B1",0,0
520 DATA "ProDel.B1",0,0
522 DATA "EpmDOS.C1",63488,2047
599 DATA "****",0,0: REM Do not touch!
600 LET n$="": LET s=0: LET l=0
610 FOR i=CODE "A" TO CODE a$
620 READ n$,s,l: REM Read in the file
name, start address and number of bytes
630 IF n$="****" THEN PRINT 'TAB 10;
FLASH 1;"Select Again!": PAUSE 100: RUN
640 NEXT i: CLS
654 REM *** Run the BASIC Program
Module
650 IF n$(LEN n$-1)="B" THEN PRINT #4:
LOAD n$
660 IF n$(LEN n$-1)<>"C" THEN CLS :
PRINT "File type error in file: ";n$:
STOP
665 REM *** Load the machine
code/screen and print the start address
and length of code.
670 IF s=0 AND l=0 THEN PRINT #4: LOAD
n$CODE
675 IF s<>0 THEN PRINT "Code located
at address: ";s: IF l<>0 THEN PRINT "to
address: ";s+l
680 IF s<>0 AND l=0 THEN PRINT #4:
LOAD n$CODE s
690 IF s<>0 AND l<>0 THEN PRINT #4:
LOAD n$CODE s,l
8999 STOP
9000 REM *** Save routine
9010 PRINT #4: SAVE "Menu.B1" LINE 1
9020 RUN
```



# T/S-1000 I/O BOARD

By Joe Rampolla

Here is how the the T/S 1000 with the I/O board is set up with my layout.

The relays on the I/O board activate separate relays which have their own power supply. The relays on the I/O board get their power from the computer. A relay is activated by POKING the appropriate value(s). All 8 relays are activated by POKING 255.

```

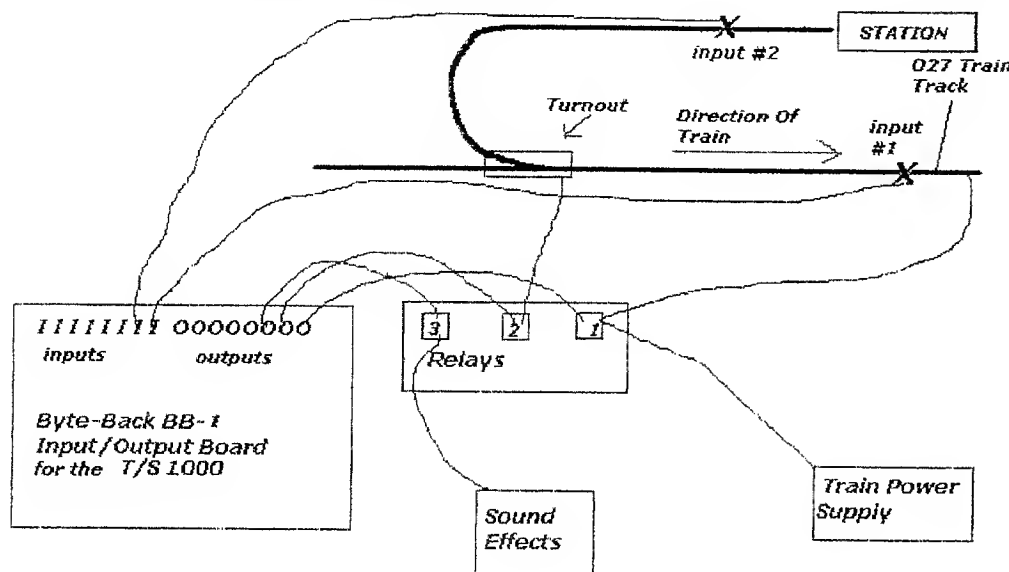
230 PAUSE 40
235 POKE X, 2
236 PAUSE 10
237 POKE X, 0
238 PAUSE 10
240 POKE X, 1
250 PAUSE 40
260 POKE X, 0
300 LET B=PEEK Y
301 IF INT (B/4) <> INT ((B/4) +
302 GOTO 300
315 POKE X, 1
320 PAUSE 40
330 POKE X, 4

```

POKE X,1 activates relay #1, breaking track current.

POKE X, 2 activates relay # 2, activating the turnout's solenoid.

POKE X, 4 turns on a train station sound effects tape by way of a tape player.



The inputs are done by way of a small light bulb causing the photo resistor to ground the input, which indicates to the computer that a particular input is "on" by way of a PEEK 16381. There is no physical connection here to the computer, just the light activates the input. The photo resistor is wired to the I/O board's input. However, in the case of the output, there is a physical connection to the computer by way of the output board's small normally open relays, however there isn't an actual electrical connection to the computer's circuitry. (In my case, an insulated rail or magnetic reed switch turns on a small light bulb to activate an input.)

The following T/S BASIC program excerpt will stop the train, throw a turnout's switch, back the train into the station and turn on a sound effects tape.

```

10 LET X = 16382
20 LET Y = 16381
90 LET B=PEEK Y
100 IF INT(B/2) <> INT ((B/2) + 0.5)
THEN GOTO 200
110 GOTO 90
200 POKE X, 1
210 PAUSE 40
220 POKE X, 0

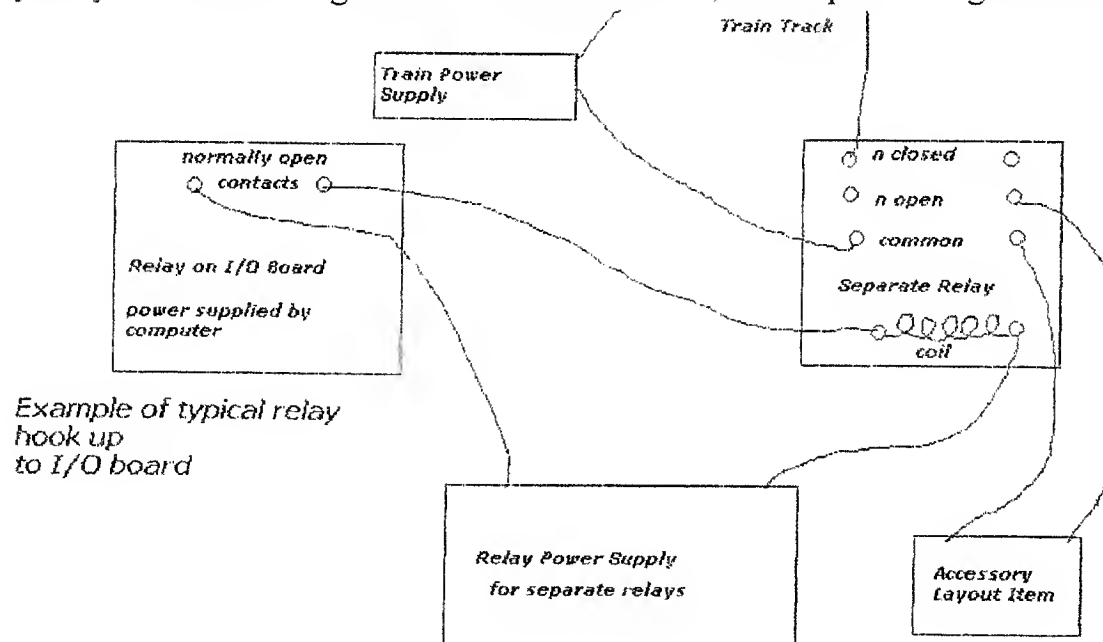
```

If PEEK Y =1, then input # 1 is grounded and "on."

If PEEK Y = 2, then input #2 is grounded.

If PEEK Y = 3, then both input #1 and #2 would be on, however this never happens when only one train is in operation.

Lines 90 - 110 are a loop waiting



for the train to hit the track section that grounds input # 1. The input byte is unpacked by line 100. Only input # 1 is being checked here. When the train reaches this section, the loop is broken and line 200 breaks the current to the track, causing the train to stop dead.

Line 220, after a pause, re-applies current to the track, causing the train to be in neutral. There is a reversing unit in the engine that cycles forward-neutral-reverse-neutral, etc., every time current is broken then re-applied. In the case of a train where polarity is reversed for reverse movement, then an additional relay is used to change polarity.

Line 235 activates the turnout's switch machine momentarily.

Line 240 removes current a second time to the track.

Line 260 re-applies current to the track and train reverses

direction.

Line 300 - 302 waits for the train to hit the section of track near the r/r station, activating input # 2. Only input #2 is being checked here.

Line 315 breaks track current, causing train to stop.

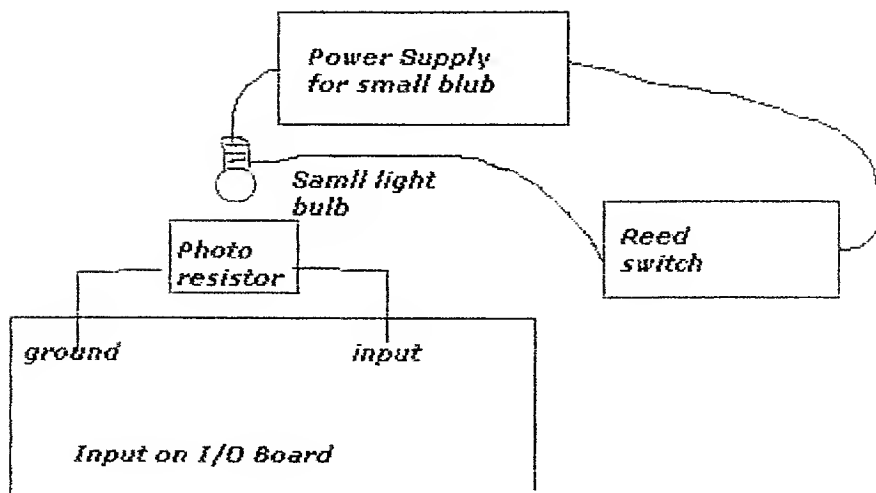
Line 330 turns on sound effects by way of POKE 4, and applies current to the rails, with train now in neutral with full interior illumination.

To activate multiple relays, simply add up the corresponding numbers. For example, if you want sound effects on all the time, POKE X, 5 would be line 200 (1 for relay #1, and 4 for relay # 3). See my input/output table. Simply add up the corresponding number to activate relays. POKE X, 36 would activate relays #6 (POKE 32) and # 3 (POKE 4). Unpacking the input byte is a little harder. Notice line 100

Simply change the value the B is divided by to check for a particular input to be "on." To check for input

#8, any number greater than 127 in PEEK 16381 would indicate input # 8 is "on." In my case, I only check for one input at a time, so I don't need to unpack the whole input byte, just one single bit. Please see the attachments In the second diagram, notice how a second device can be activated by one relay. It is best to use 4PDT relays if possible. A relay can do double,

even triple duty, by having different track sections wired to just one relay, and its function would change depending on

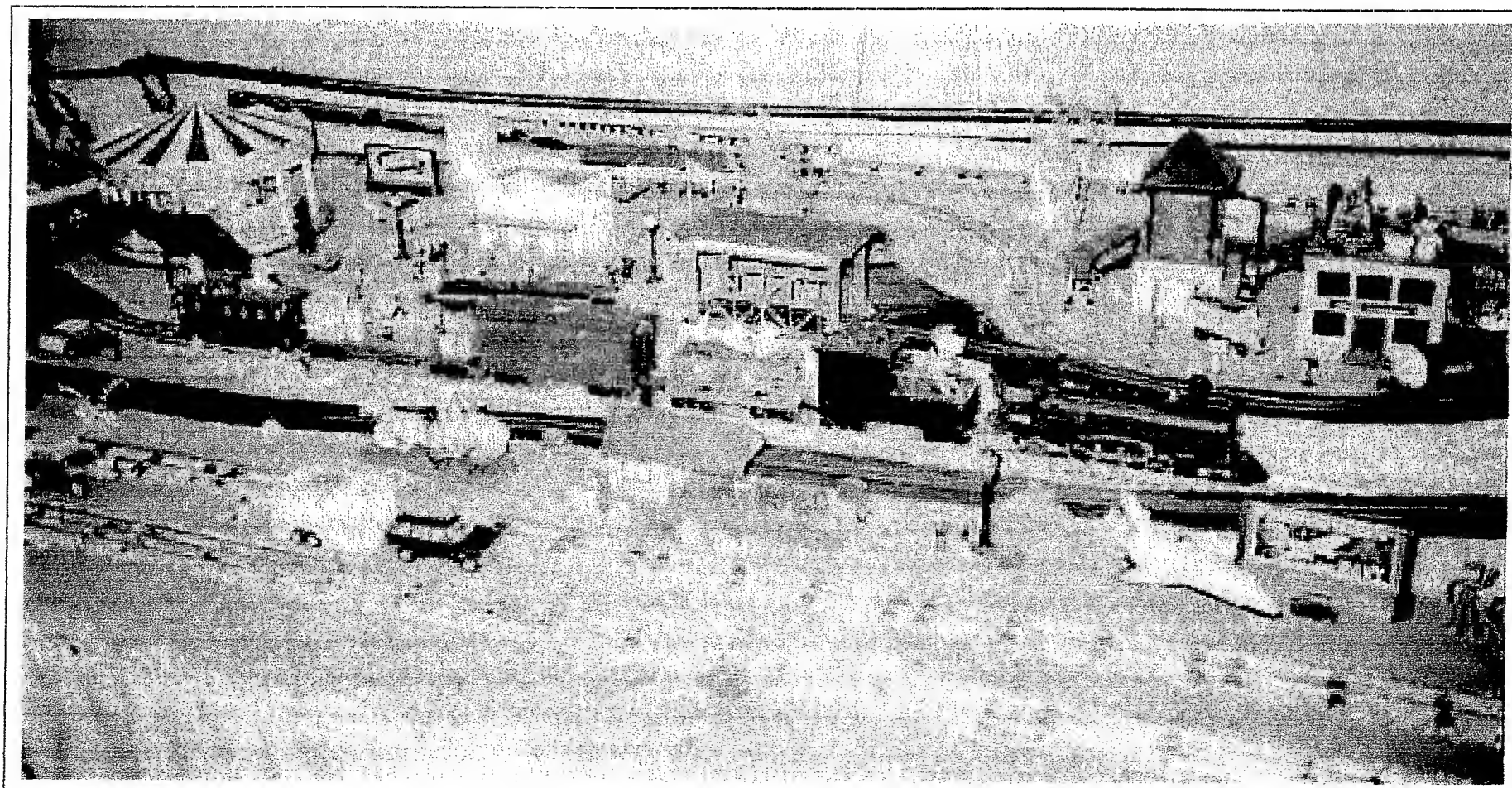


RELAY	#8	#7	#6	#5	#4	#3	#2	#1	
OUTPUT	POKE 128	POKE 64	POKE 32	POKE 16	POKE 8	POKE 4	POKE 2	POKE 1	POKE 16382
INPUT	BYTE/ 2 # 1	BYTE/ 4 # 2	BYTE/ 8 # 3	BYTE/ 16 # 4	BYTE/ 32 # 5	BYTE/ 64 # 6	BYTE/ 128 # 7	>127 # 8	PEEK 16381

Unpack Byte: IF INT (B/V) <> ((B/V) + 0.5) THEN etc. Btve/V above.

#### QUICK REFERENCE INPUT/OUTPUT TABLE

where the train may be at any given time.



# Unclassified Ads

**Place your ads here, it is free!**

Mail/E-mail to: A. KAHLE 3343 S FLAT ROCK CT SIERRA VISTA AZ 85650-6874

**Please inform and/or update the Editor of any changes in your ad/s**

## Home Electronics Service

We have been a part of the Sinclair scene since 1982, repairing  
**ZX Spectrums** for Sinclair Research in England.

We provide Sales, Service, and Software for the

**QL, Spectrum, ZX-81 and Z88.**

[www.members.tripod.com/hes\\_computing/hes1.html](http://www.members.tripod.com/hes_computing/hes1.html)

E-Mail 74601.1535@compuserve.com

Hours of Operation is Monday - Friday 1300 hrs. to 2100  
hrs. central time zone.

Phone 210 661-4376

**John R. Rish**

Home Electronics Service

5222 Kazen Dr.

San Antonio TX 78219 USA

Bill Mathis

## Pro Digital Electronics

323 SE 28th Ave.

Portland, Oregon 97214

503-232-3200

## Pro Act Consulting, Inc.

2660 N. Houghton Rd.

Tucson, AZ 85749

520-749-5395, fax 520-749-3626

email <proactmd@aol.com>

## The John Oliger Co.

11601 Widbey Dr.

Cumberland IN 46229

**The John Oliger Floppy Disk System**

**FOR THE TS-2068**

EXPANSION BOARD

2068 User Cartridge

DISK BOARDS "A" & "B"

2068 Parallel Printer Port

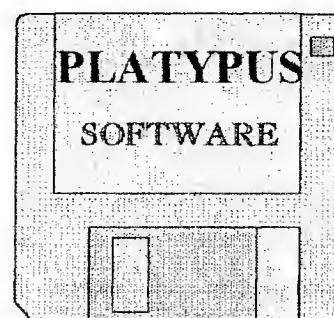
2068 EPROM Programmer

2068/SPECTRUM Joystick Port

DFh Mapped Universal I/O Port board

User Manual only : \$5.00 (Read before you buy)

[joliger@mindspring.com](mailto:joliger@mindspring.com)



QLAMBer \$20

QLuMSi \$20

SeekQL \$10

Upgrades \$5

*Al Feng*

914 Rio Vista Cir SW

Albuquerque NM 87105

(505) 843-8414

## Make David Leech an Offer

ZX-81/TS-1000 TS-2068 Hardware Kits

Real Time Clock I/O Controller RS-232

Centronics I/F 16K & 64K RAM 300 BAUD

Modem A-D Converter(assembled)

**BYTE-BACK INC**

536 Long Ter

Leesville SC 29070

**ZX-TEXT** - Word Processor

**ZX-CALC** - Spread Sheet

**Business Software**

**Cycle Accounting** Financial Report Generator

**ZX-CALENDAR** - Time Management

**ZX-81 TS-1000 TS-1500 TS-2068**

Albert F. Rodriguez

**A.F.R. Software®**

1605 Pennsylvania Ave 204

Miami Beach FL 33139

305 531-6464

**WOOD & WIND**

## ARCHIVE Based QL Software

**QLerk** - A complete financial program for the QL

QLerk software (v3.21) with tutorial \$29

QLerk manual \$29

QLerk software & manual \$50

**DBEasy** - A menu based database system

DBEasy software (v1.6) \$24

DBEasy upgrade from V1.5 \$7

**DBProgs** - A toolkit of ARCHIVE procedures

DBProgs software (v1.8) \$18



DBProgs upgrade from V1.7 \$7  
DBTutor - A general purpose learning program  
DBTutor software(v1.5) \$12  
PC DBEasy - Just like QL DBEasy  
PC DBEasy software (v1.3) \$12

Bill Cable

### Wood & Wind Computing

RR3 BOX 92  
Cornish NH 03745 USA  
Phone (603) 675-2218

## QBox USA

810 254-9878

24 hours a day  
300 to 28,800 bps

Supporting all Sinclair and Timex users Fidonet Echomail arefor Sinclair computers. Lots of new files for you to download such as TS-2068 emulator for those who use a PC give us a call and let us know what you want to see. Message and file areas. QL International, Quanta, QL Hacker's Journal, Spectrum/2068, ZX-81/TS-1000, Z88, NetMail, emulators, pointer, FDFormat for QXL/QDOS, etc.

**Sysop John J Impellizzeri**  
**Co-Sysop Don Waltermann**  
**Utica, Michigan, USA**

## Domino Cubes Z88

Hardware & Software

352 7<sup>th</sup> Ave. 15<sup>th</sup> Flr.

New York, NY 10001

Phone 212 631-7563

Fax 212 947-5069 Voice mail pager 917 490-8407

Domino.cubes@excelsior.net

## LISTing Newsletter

The Long Island Sinclair/Timex Users Group

Robert Malloy, Treasurer

412 Pacific St.

Massapequa Park, NY 11762

## NESQLUG NEWS

New England Sinclair QL Users Group

Ed Kingsley, Editor

16 Highland Avenue

Saugus MA 01906

(781) 233-3671 EdK4@aol.com

# QL Hacker's Journal

Supporting All QL Programmers

Timothy Swenson, *Editor*

2455 Medallion Dr.

Union City, CA 94587-1914

swensontc@geocities.com

<http://www.geocities.com/SilconValley/Pines/5865/>



**Newsletter**

**The Capital Area T/S Users Group**

Barry Washington

7044 Cindy Ln

Annandale VA 22003

301 589-7407

BBS 301 588-0579

Internet mf0002@epfl2.epflbalto.org

Alec Carswell

## Motivation

16 Montgomery Ave

Beith Ayrshire KA15 1EL

United Kingdom

## The Ramtop

*Newsletter*

The Greater Cleveland T-S User Group

Thomas Simon Editor

615 School Ave

Cuyahoga Falls OH 44221

E-Mail CIS 73177,333

Jon Kaczor Production

4568 Williamson Ave

Brooklyn OH 44144

75363.1127@compuserve.com

## ZX-TEAM MAGAZIN

Peter Liebert-Adelt

LUETZOW STR 3

D-38102 BRAUNSCHWEIG

GERMANY

Email: p.liebert@t-online.de

<http://home.t-online.de/home/p.liebert/zx-team.htm>

Amateur Radio: DK4BF@DB0FC.#NDS.DEU.EU

**The ZX Spectrum 48/128 Emulator  
for IBM & Compatibles: Z80 Version**

Turn your PC into a real ZX Spectrum 48/128

⇒ Full Spectrum emulation, border, flash, beeper, Interface 1,

Microdrive in cartridge file, RS232 input and output redirection to file, COM or LPT, joystick support, 128K sound through Soundblaster or internal speaker, built-in monitor,

⇒ Able to load ANY, even protected or speed-saved program from tape, to save to tape, to redirect tape loads and saves to disk for easy file access,

⇒ 2500 line English documentation.

⇒ Runs okay under DOS, Windows and DesqView,  
- Full source code of emulator and utilities included!

Runs on any 640K PC; too slow for practical use on PC/XT's but fast enough on AT's, uses VGA/EGA/CGA or Hercules.

This program costs US \$20. You will receive a 3.5" DD disk (5.25" disks on request), and you'll be kept informed about updates. Please send bank notes (bills), name and address to:

### **Gerton Lunter**

PO Box 2535  
NL-9704 CM Groningen  
Netherland

If you send a cheque, please add US \$15 extra and allow 4 weeks for delivery.

## **Jochen Merz Software**

### **The ROMDisk Fully Functional**

Extremely small board for the QL's ROM-Port, works like a permanent RAM disk.

In order to be able to upgrade to the color drivers, you need to have SMSQ/E. The *normal* SMSQ which is shipped with QXL card will not be able to handle more colors.

### **SMSQ/E for the QXL**

As Aurora owners will be able to use more colors when the color drivers are available, another offer for SMSQ/E for the Aurora plus GoldCard/Super/GoldCard:

### **SMSQ/E for the Super GoldCard**

**QL Games & Upgrades QL Applications**

**ProWesS + Applications**

Jochen Merz Software

Im stillen Winkel 12

47169 Duisburg, Germany

☎ 0203-502011 Fax 0203-502012

**Credit Cards accepted**

<http://www.j-m-s.com/smsq/>

e-mail [smsq@j-m-s.com](mailto:smsq@j-m-s.com)

# **QL Today**

QL Today is published by Jochen Merz Software. Jochen Merz has been supplying software for the QL for several years and has built up a good reputation for quality and fair trading. The representative in Britain is Miracle Systems Ltd. who take subscriptions and do the distribution.

### **Subscriptions**

Germany (+German add-on) DM 70

England DM 60 £25

Rest of the world DM 70 £30

Back-issues are available for DM 12 (incl. postage)

Checks should be made payable Miracle Systems Ltd.

### **English Office**

Miracle Systems Ltd.

20 Mow Barton

Yates, Bristol

United Kingdom BS17 5NF

Tel. +44 1454 883602 Fax. +44 1454 883602

### **Editor**

Dilwyn Jones

41 Bro Emrys

Tal-Y-Bont, Bangor, Gwynedd

United Kingdom LL57 3YT

Tel. +44 1248 354023 Fax. +44 1248 354023

## **Items for the Timex\Sinclair Computer**

**Timeworks Programming kit #1** For T/S 1000 & ZX81 \$4.95

**Mindware Gulp Game** Timex 1000 & Sinclair ZX81 \$4.95

**Timex Horace & The Spiders** for the 2068. \$5.95

Timex Sinclair 1000 software on tape

**Chess** (16K RAM) qty 5 price \$2.95 ea

**The Challenger I** qty 17 price \$2.95 ea

**Mindware MW100** plain paper printer for the TS1000 or ZX81 computer made in the US. \$45.00

All items are new. Please add \$3.95 shipping to all orders.

MC, VISA, American Express. phone 717-748-1747

### **Keith Electronics**

224 North Grove St

Lock Haven, PA. 17745

## **ZX-81 Kit**

**\$30 Plus \$10 for S/H**

**TS-2040 Printers 16K RAM**

We Do Not Ship Outside The US & Canada

Checks, Money Orders and Visa or Mastercard

Call 212-675-8414 FAX 212-675-8980

**Zebra Systems, Inc.**

122 W26th St. Suite. 904

New York, NY 10001

Shipping is by UPS ground.

**Basics of Timex Sinclair 1500/1000 BASIC**

Allen Wolach, Separately priced \$12.95

**BASIC Basics for the Timex/Sinclair 1500/1000**

Michael Barnett/Simon Barnett, Separately priced \$12.95

**The Ins and Outs of the Timex TS-1000 & ZX-81**

Don Thomasson, \$12.95

**Computer Interfacing Technique in Science TS-1500/1000**

Paul Field & John Davies, \$12.95

**ZX-81 BASIC Programming (in Spanish language)**

Steve Vickers (Traducido y adaptado) A copy of this book is shipped free for asking if you just pay for additional shipping cost. Separately priced \$10

## **Sinclair Resources**

John McMichael (Developer - Graphics)

1710 Palmer Dr

Laramie WY 82070

Keith Watson (AERCO & Z80 Emulator)

41634 Amberly Dr.

Mt. Clemens, MI 48038

Rod Gowen (RMG)  
14784 S Quail Grove Cir  
Oregon City OR 97045-8843

Send them a LSASE and ask for information about  
their current products and/or services.

**EPS**  
**EPROMS** AS LOW AS **25¢**

27C256	27C040	27C1024	87C51
27C512	27C400	28F010	8751
2732A	27C4096	28F020	8755
27C32	574000	28F040	MC68HC70
27C128	574200	8749H	5C8A
27C84	27C402	8748H	PIC18C56
2716	27C020	8741	PIC16C54
2708	27C210	8742	PIC18C622
27C512-90	27C010	8744	GAL16V8

**STATIC RAM**  
HM628128  
62256  
6264  
6116

**PROCESSOR**  
8255  
8254  
80C31

• Many more parts in stock  
• All major brands  
• All guaranteed  
E-Mail: [eproms@aol.com](mailto:eproms@aol.com)

TEL: (818) 774-9444 · FAX: (818) 774-0822  
**WE BUY EXCESS INVENTORY**

# Surplus T/S Inventory

JOHN J SHEPARD III  
281 130<sup>th</sup> ST  
OGDEN IA 50212  
< [jshepard@netins.net](mailto:jshepard@netins.net) >  
Mostly QL & TS-2068

JACK BOATWRIGHT  
67325 FRYREAR RD  
BEND OR 97701  
< [jboatno4@outlawnet.com](mailto:jboatno4@outlawnet.com) >  
Mostly ZX-81/TS-1000 & TS-2068

**QL TS-2068 ZX-81**  
**Books Magazines**  
**Software**

*Joe Rampolla's train*

